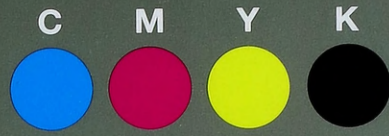
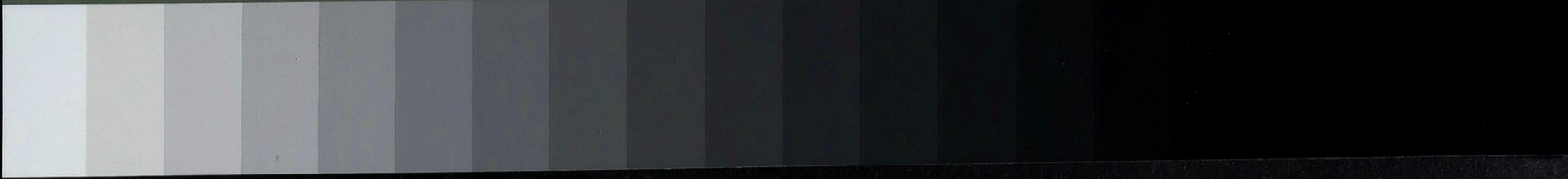


Grey Scale #13



DANES-PICTA.COM

A 1 2 3 4 5 6 M 8 9 10 11 12 13 14 15 B 17 18 19



AKADEMIA OBRONY NARODOWEJ

CENTRUM INFORMATYKI

KATEDRA LOGISTYKI WYDZIAŁU WOJSK LĄDOWYCH

JAWNE



ZASTRZEŻONE

Egz. Nr 1..



ZAPATRYWANIE W MPS BRYGADY
ZMECHANIZOWANEJ WYKONUJĄCEJ
MARSZ



60782

WARSZAWA

1995



Kędziorek

AKADEMIA OBRONY NARODOWEJ

CENTRUM INFORMATYKI
KATEDRA LOGISTYKI WYDZIAŁU WOJSK LĄDOWYCH
JAWNE

*160306 Anna KOLEK
Podst. pnel. NV uch. 648
z dn. 24.02.2006*

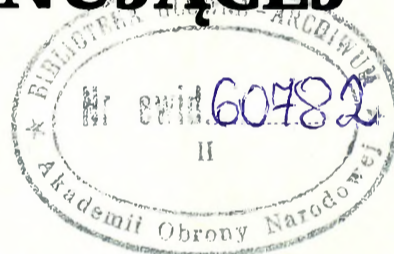


**ZASTRZEŻONE
POUFNE**

Egz. nr 1...



ZAOPATRYWANIE W MPS BRYGADY ZMECHANIZOWANEJ WYKONUJĄCEJ MARSZ



*Prochownik
podany plan na ZASTRZEŻONE
3.1.01-2013 Jan...*

MPS

29

7.17

Opracował zespół w składzie:

płk dr inż. Zbigniew KLIMKIEWICZ
płk dr Aleksander ŁOJKO

S P I S T R E Ś C I

WPROWADZENIE	4
1. OGÓLNA CHARAKTERYSTYKA MARSZU ORAZ WYNIKAJĄCE Z NIEJ WŁAŚCIWOŚCI ZABEZPIECZENIA W MATERIAŁY PĘDNE I SMARY	6
1.1. Wpływ warunków marszu na proces zabezpieczenia wojsk w MPS...	9
2. ZABEZPIECZENIE MATERIAŁOWE BATALIONU I BRYGADY W MARSZU	12
3. UWARUNKOWANIA TAKTYCZNO-LOGISTYCZNE MAJĄCE WPŁYW NA WIELKOŚĆ POTRZEB PALIW PŁYNNYCH W MARSZU	14
4. METODY OKREŚLANIA ZUŻYCIA MPS W MARSZU	17
4.1. Metoda statystyczna.....	17
4.2. Metoda matematyczna.....	17
4.2.1. Współczynnik eksploatacyjny k_e	18
4.2.2. Współczynnik manewrowy k_{mt}	19
4.3. Obliczanie zużycia MPS przez poszczególne kolumny.....	20
5. OPIS PROGRAMU "ZAOPATRYWANIE W MPS BRYGADY ZMECHANIZOWANEJ W MARSZU" (MPSMARSZ)	22
5.1. Przeznaczenie programu.....	22
5.1.1. Język programowania.....	22
5.1.2. Sprzęt.....	22
5.1.3. Pliki systemu.....	23
5.2. Uruchomienie programu.....	23
5.3. Obsługa programu.....	24
5.3.1. Zestawienie masy jednostek napełnienia pododdziałów.....	26
5.3.2. Sformowanie kolumny marszowej.....	27
5.3.2.1. Sformowanie nowej kolumny.....	28

5.3.2.2.	Przeglądanie istniejących kolumn.....	30
5.3.3.	Obliczanie masy jednostek napełnienia kolumn marszowych.....	31
5.3.4.	Obliczanie zużycia MPS przez poszczególne kolumny.....	31
5.3.5.	Podział posiadanych sił i środków służby MPS.....	35
5.3.5.1.	Wprowadzenie danych o siłach i środkach służby MPS.....	35
5.3.5.2.	Wyświetlanie danych.....	36
5.3.5.3.	Podział sił i środków służby MPS.....	38
5.3.6.	Operacje dyskowe.....	41
5.3.6.1.	Odczyt tabeli jednostek napełnienia.....	42
5.3.6.2.	Odczyt składu kolumn.....	43
5.3.6.3.	Odczyt środków służby MPS.....	44
5.3.7.	Zakończenie pracy programu.....	46

Załącznik

1. Postać źródłowa programu

W P R O W A D Z E N I E

Przegrupowanie wojsk jest nieodłączną częścią składową każdego działania bojowego i może być wykonywane zarówno przed wybuchem wojny, w czasie narastania zagrożenia wojennego, jak i w toku trwania działań wojennych.

We współczesnych warunkach działań bojowych wojska mogą przemieszczać się na setki kilometrów i pokonywać marszem zachowując jednocześnie pełną zdolność do walki.

"M a r s z" jest to zorganizowane przemieszczenie wojsk w kolumnach na własnych środkach transportu do wyznaczonego rejonu lub na rubież walki, z zachowaniem zdolności do wykonywania zadań bojowych^{*)}.

Związki taktyczne i oddziały mogą wchodzić do walki z marszu lub po krótkim pobycie w rejonie wyjściowym.

Zachodzi więc konieczność dokładnego zaplanowania i ścisłego realizowania zadań zabezpieczenia materiałowego wojsk, zwłaszcza zaopatrzenia w materiały pędne i smary (MPS), gdyż ono to głównie decyduje o sprawnym przebiegu marszu.

Głównym czynnikiem decydującym o organizacji i pracy organów logistycznych w czasie przygotowania do marszu, jak i podczas jego trwania jest czas. Współcześnie przygotowanie wojsk do marszu odbywać się będzie w ograniczonym czasie, dlatego należy dążyć do maksymalnego

^{*)} *"Regulamin działań taktycznych wojsk lądowych.
Część I. (ZT - oddział) str. 56. Wyd. Sztab Generalny
Warszawa 1994r*

wykorzystania istniejącej techniki komputerowej do usprawnienia prac kalkulacyjno-planistycznych w tym zakresie.

Prezentowane opracowanie pozwala na usprawnienie prac planistyczno-kalkulacyjnych w zakresie zaopatrywania w MPS oddziału (BZ) wykonującego marsz.

W opracowaniu przedstawiono:

- ogólną charakterystykę marszu oraz wynikające z niej właściwości zabezpieczenia w materiały pędne i smary;
- zabezpieczenie materiałowe batalionu i brygady w marszu;
- uwarunkowania taktyczno-logistyczne mające wpływ na wielkość zużycia paliw płynnych w marszu;
- metody określania zużycia MPS w marszu;
- podstawowe wzory do obliczania parametrów operacyjno-logistycznych w zakresie MPS;
- opis wykorzystania programu komputerowego.

1. OGÓLNA CHARAKTERYSTYKA MARSZU ORAZ WYNIKAJĄCE Z NIEJ WŁAŚCIWOŚCI ZABEZPIECZENIA W MATERIAŁY PĘDNE I SMARY

Każde działanie bojowe poprzedza odpowiednie przemieszczenie wojsk. Może być ono realizowane zarówno w skali operacyjnej, jak i taktycznej.

Przed wybuchem wojny oraz w jej początkowym okresie a także w czasie jej trwania wojska przemieszcza się do rejonu ześrodkowania. W rejonach tych, w celu utworzenia odpowiedniego ugrupowania operacyjnego czy taktycznego dla wykonania określonego zadania bojowego, dokonywane będą przemieszczenia oddziałów i pododdziałów, zarówno w kierunku frontu (czołowe, bokowe, boczne), czy też odfrontowe. Niekiedy takie przemieszczenie może być dokonywane z rejonu alarmowego lub garnizonu.

Przemieszczenie oddziałów i pododdziałów może być dokonywane na własnych środkach transportu czy sposobem kombinowanym. Mogą też wojska być przewożone transportem kolejowym, wodnym lub powietrznym.

Głównym sposobem przemieszczania wojsk będzie jednak marsz.

Związek taktyczny (oddział) najczęściej przemieszcza się na własnych środkach transportu, co zapewnia:

- utrzymanie zwartości oraz gotowości bojowej podległych oddziałów i pododdziałów do wykonania zadań

- zmniejsza ich wrażliwość na uderzenia przeciwnika;
- ułatwia manewr;
- a także pozwala uzyskać odpowiednie tempo marszu i zajmując zaplanowany rejon w ugrupowaniu dogodnym do walki.

Wojska mogą wykonywać marsz w celu:

- ześrodkowania w wyznaczonym rejonie;
- opanowania dogodnej rubieży;
- dokonania przegrupowania sił i środków na zagrożony kierunek;
- zniszczenia wysadzonego desantu lub okrążonego przeciwnika.

Możliwość przegrupowania ZT (oddziału) na etatowych pojazdach mechanicznych określa głównie odległość, jaką mogą przebyć wojska w ciągu 10-12 godzin.

W ciągu doby odległość marszu może wynosić:

- kolumn mieszanych i czołgowych do 300 km;
- kolumn samochodowych do 400 km.

Średnia prędkość marszu, bez uwzględniania czasu na odpoczynki może wynosić dla:

- kolumn mieszanych i gąsienicowych 25-30 km/godz;
- kolumn samochodowych 30-40 km/godz i więcej.

W górach, terenie lesisto - jeziornym i innych niesprzyjających warunkach średnia prędkość marszu, w zależności od charakteru terenu i stanu dróg, może się zmniejszyć do 20 km/godz, a długość marszu dobowego może wynosić 200-250 km, a niekiedy nawet i mniej.

Dla regeneracji sił żołnierzy, spożycia posiłków, uzupeł-

nienia paliw, sprawdzenia stanu uzbrojenia, sprzętu wojskowego oraz obsługę techniczną pojazdów wyznacza się postoje i odpoczynki.

Postój jednogodzinny wyznacza się po kilku godzinach marszu a postoje dwugodzinne w drugiej połowie marszu dobowego. Po wykonaniu marszu dobowego wyznacza się odpoczynek dzienny (nocny).

Od liczby postojów i odpoczynków oraz czasu ich trwania / zależy organizacja i praca sił i środków logistycznych w zakresie zabezpieczenia logistycznego przemieszczających się wojsk.

Batalion wykonuje marsz w składzie brygady a niekiedy samodzielnie. W składzie brygady może on maszerować:

- w kolumnie sił głównych;
- stanowić ubezpieczenie czołowe (awangardę) lub ubezpieczenie tylne.

Batalionowi wyznacza się jedną drogę marszu.

Brygada wykonuje marsz w zasadzie w składzie ZT, może także przemieszczać się samodzielnie. W składzie ZT może ona wykonywać marsz w kolumnie sił głównych lub stanowić oddział wydzielony. Brygadzie wyznacza się zazwyczaj jedną drogę marszu. Natomiast gdy działa ona w roli oddziału wydzielonego to wyznacza się drogę (drogi) marszu, która nie powinna przebiegać przez duże miejscowości, węzły kolejowe itp. mogące być celami uderzeń lotnictwa przeciwnika.

Od liczby i stanu otrzymanych do wykorzystania dróg oraz długości kolumn marszowych zależy miejsce poszczególnych podod-

działów (oddziałów) logistycznych w ugrupowaniu marszowym.

Oddziały i pododdziały logistyczne maszerują samodzielnie kolumnami za siłami głównymi maszerujących wojsk.

Oprócz wymienionych operacyjno-taktycznych warunków prowadzenia marszu na organizację i pracę sił i środków logistycznych brygady i batalionu będą mieć wpływ również te czynniki, które występują w każdym warunkach działań wojsk tj.:

- aktywność lotnictwa;
- warunki terenowe i atmosferyczne;
- pora roku i doby;
- możliwości przeciwnika w zakresie minowania powierzchniowego dróg marszu;
- działania desantów i grup dywersyjno - rozpoznawczych itp.

1.1. Wpływ warunków marszu na proces zabezpieczenia wojsk w MPS

Zabezpieczenie wojsk w MPS ma istotny wpływ na terminowość i płynność ruchu wojsk oraz utrzymania ich w wysokiej zdolności bojowej podczas marszu.

Do podstawowych warunków wykonywania marszu, mających bezpośredni wpływ na proces zabezpieczenia wojsk w czasie marszu w MPS należy zaliczyć:

- stopień zagrożenia ze strony przeciwnika naziemnego i powietrznego;
- długość i tempo marszu;

- ~~długość i tempo marszu;~~
- sprawność techniczna sprzętu wojskowego;
- wyszkolenia kierowców pojazdów i wozów bojowych;
- warunki geofizyczne.

W warunkach braku oddziaływania przeciwnika lub gdy oddziaływanie to jest niewielkie, marsz będzie wykonywany w sposób płynny. Ruch wojsk będzie zgodny z wcześniej opracowanym planem, bez żadnych zbędnych zatrzymań. Może tylko wystąpić konieczność wykonywania objazdów dróg marszu a także powstanie minimalnych strat w ludziach i sprzęcie wojskowym.

Oddziaływanie przeciwnika bronią o różnym zasięgu i możliwościach rażenia powoduje znaczne utrudnienie, zarówno dla maszerujących wojsk, jak i zaopatrzenia ich w paliwa płynne.

Zakłócenia spowodowane tym oddziaływaniem prowadzą do korekt planów przegrupowania, zmiany dróg marszu, przepraw, rejonów odpoczynków itp. Powoduje to potrzebę wyznaczania innych źródeł zaopatrzenia. Oddalenie tych źródeł zaopatrzenia od dróg marszu oraz możliwości dystrybucyjne mogą wpłynąć w sposób istotny na czas odtworzenia zapasów MPS w pododdziałach zaopatrzenia maszerujących wojsk.

Oddziaływanie przeciwnika może również spowodować straty w środkach dystrybucyjno - paliwowych oddziałów (pododdziałów) zaopatrzenia, a także w wydzielonych źródłach zaopatrzenia w MPS.

Na zużycie MPS wpływ takich elementów, jak długość marszu i warunki geofizyczne wydają się na tyle oczywiste, że zrezygnowano z ich opisywania w prezentowanej pracy.

W czasie przemieszczania wojsk na terenie macierzystego

okręgu wojskowego (OW) proces zabezpieczenia ich w paliwa płynne będzie realizowany nie tylko siłami i środkami maszerujących wojsk ale siłami i środkami gospodarki narodowej i układu pozamilitarnego.

Najgorsza sytuacja wystąpi w zaopatrywaniu w MPS wojsk, które będą przemieszczały się na terenie innego okręgu wojskowego. W takim przypadku już w czasie pokoju muszą być określone źródła zaopatrzenia, wydzielone rezerwy materiałowe, a także możliwe warianty sposobów ich podjęcia przez te wojska na terenie innego OW.

2. ZABEZPIECZENIE MATERIAŁOWE BATALIONU I PUŁKU W MARSZU.

Jak wcześniej wspomniano podstawowym środkiem materiałowym zużywanym przez wojska w czasie marszu będą paliwa płynne.

Priorytetowym zadaniem dowódców, sztabów i organów logistycznych jest zaopatrywanie wojsk w paliwa do wielkości określonych normami zaopatrzenia zarówno w okresie przygotowania jak i w toku marszu.

W okresie przygotowania do marszu (jako zasada) należy zgromadzić pełne zapasy ruchome paliw płynnych na wszystkich szczeblach organizacyjnych wojsk w ilości określonej normami, a także odpowiednie zapasy doraźne.

Urzutowanie zapasów ruchomych MPS szczebla taktycznego przedstawiono w poniższej tabeli.

Tabela 1

Urzutowanie zapasów ruchomych MPS (jn*)

Rodzaj MPS	Urzutowanie	Przy sprzęcie	W pododdziałach logistycznych batal.	W pododdziałach logistycznych oddziału	W oddziale logistycznym ZT	Razem w ZT
Benzyna samochodowa		5	1	2	2	10
Olej napędowy:						
- do pojazdów kołowych		5	1	2	2	10
- do pojazdów gąsienicowych		3	1	2	2	8

*) *Jednostka napełnienia (jn) odnosi się do pojedynczego pojazdu mechanicznego i określa normę MPS zużywaną przez niego do przebycia 100 km po drodze bitej, agregat w ciągu jednej godziny pracy lub samolot (śmigłowiec) w ciągu jednej godziny lotu.*

Zapasy doraźne paliw wynoszą:

- benzyny samochodowej i oleju napędowego do pojazdów kołowych - 1.5 jn;
- oleju napędowego do pojazdów gąsienicowych 0.6 - 1.5 jn;.

Zużycie paliw w czasie marszu zależy od takich czynników jak:

- odległość drogi marszu;
- czas trwania marszu;
- pora roku;
- warunki atmosferyczne i drogowe;
- stan techniczny pojazdów;
- inne.

Głównym jednak czynnikiem jest długość drogi marszu.

Wykorzystując ćwiczenia z wojskami wyznaczono metodą empiryczną średnie zużycie MPS podczas marszu na odległość 100 km.

Tabela 2

Wielkość zużycia paliw na 100 km marszu (jn)

Oddział	Pobyt w rejonie		Marsz	
	BS i ON do poj. kołowych	ON do poj. gąsienicowych	BS i ON do poj. kołowych	ON do poj. gąsienicowych
ZT	0.5	0.45	1.5 - 1.60	1.5 - 1.65
BZ	0.5	0.6	1.5 - 1.60	1.5 - 1.65

W warunkach zimowych, przy niskich temperaturach (poniżej -5°C) i pokrywie śnieżnej $> 10\text{cm}$, zużycie wzrasta nawet do 50%.

Przy wykonywaniu marszu w terenie górzystym i bagienno-lesistym zużycie wzrasta 1.5 - 2 krotnie.

3. UWARUNKOWANIA TAKTYCZNO - LOGISTYCZNE MAJĄCE WPŁYW NA WIELKOŚĆ POTRZEB PALIW PŁYNNYCH W MARSZU

W czasie planowania marszu na dużą odległość, potrzeby paliw płynnych brygady na czas przesunięcia ustala się na podstawie stanu zaopatrzenia w chwili otrzymania zadania, nakazanych przez przełożonego wielkości zapasów rzeczowych i doraźnych paliw płynnych na początek marszu, przewidywanego zużycia w okresie przygotowania i wykonywania przesunięcia oraz wielkości paliw jakie brygada powinna posiadać po wykonaniu marszu*/ .

W okresie poprzedzającym marsz w pododdziałach brygady uzupełnia się paliwa płynne do pełnych norm zapasów ruchomych, a także gromadzi się zapasy doraźne w wielkości:

- benzyny samochodowej - 1.5jn;
- oleju napędowego do pojazdów kołowych - 1.5jn;
- oleju napędowego do pojazdów gąsienicowych - 0.6 - 1.5jn.

W marszu wykonywanym własnymi środkami transportu (na kołach) wielkość zużycia paliw płynnych zależy przede wszystkim od jego odległości. Jednak na wielkość zużycia paliw płynnych mają również wpływ takie czynniki jak: stopień intensywności oddziaływania przeciwnika, warunki atmosferyczne, tempo marszu, warunki drogowe, pora roku, stan techniczny sprzętu itp.

W przeciętnych warunkach terenowych i atmosferycznych w marszu dobowym**/ zużycie paliw może wynosić:

- benzyny samochodowej - 4.5 - 4.8jn (tj. 17 - 18t);

*/ Po wykonaniu marszu do nakazanego rejonu lub rubieży nie odtwarza się zapasów doraźnych paliw płynnych, chyba, że przełożony nakaże inaczej.

**/ Jako marsz dobowy przyjęto odległość 300km.

- oleju napędowego do pojazdów kołowych - 4.5 - 4.8jn
(tj. 12.6 - 13.5t);
- oleju napędowego do pojazdów gąsienicowych - 4.5 - 4.8jn[?]
(tj. 65 - 80t).

W warunkach zimowych, zwłaszcza przy obniżeniu temperatury poniżej minus 5°C i występującej pokrywie śnieżnej (powyżej 10cm), zużycie paliw może wzrosnąć o 50% normy, a w marszu w terenie górzystym i leśnisto-jeziornym - nawet o 50 - 100% normy. Zwiększone potrzeby paliw wystąpią również w warunkach uderzeń przeciwnika na maszerujące kolumny oraz obiekty drogowe. Zużycie w tym wypadku może wzrosnąć nawet o 30% normy z powodu objazdów i obejść stref skażeń i zniszczeń.

• Podczas marszu brygady uzupełnianie paliw w zbiornikach pojazdów mechanicznych i wozach bojowych dokonywane jest sukcesywnie w czasie postojów i odpoczynków. Jest ono koniecznym zwłaszcza w pojazdach gąsienicowych, w których zapas paliw nie wystarcza na wykonanie marszu dobowego.

W zależności od odległości marszu, przewidywanego zużycia paliw, ilości i czasu trwania postojów oraz odpoczynków, oddalenia składów MPS od dróg marszu, uzupełnianie i odtwarzanie zapasów paliw płynnych może być organizowane następującymi sposobami:

- **Sposób pierwszy** - transport z paliwami z kompanii zaopatrzenia brygady oraz z batalionu zaopatrzenia dywizji przydzielony jest do pododdziałów bojowych. Maszeruje on w ich ugrupowaniu i używany jest do uzupełnienia paliw w zbiornikach pojazdów. Opróżnione środki transportowo-dystrybucyjne do przewozu paliw kierowane są do wyznaczonych składów, gdzie odtwarzają zapasy a następnie wracają do za-

bezpieczanego pododdziału. Jeżeli podczas marszu zapasy paliw przewożone w pododdziałach zaopatrzenia nie były wykorzystywane do uzupełnienia paliw w zbiornikach pojazdów, wówczas wykorzystuje się je w całości do tankowania pojazdów w rejonie odpoczynku dziennego;

- **Sposób drugi** - transport brygady (dywizji) z paliwami kieruje się z góry do planowanych rejonów postoju lub odpoczynku, gdzie w miarę przybywania kolumn marszowych organizuje się tankowanie pojazdów;

- **Sposób trzeci** - tankowanie pojazdów przy użyciu kompanii masowego tankowania (kmt). Kompania może działać na korzyść maszerującego brygady rozwijając w określonym rejonie tzw. front wydawczy paliw. W czasie marszu ma to najczęściej miejsce w rejonach odpoczynku. Taki sposób tankowania pojazdów umożliwia kontynuowanie marszu bez naruszania zapasów paliw przewożonych w pododdziałach zaopatrzenia.

Źródłami uzupełnienia paliw dla maszerującej brygady w marszu na dużą odległość mogą być: stacjonarne składy MPS wojskowe i cywilne, transporty kolejowe z paliwami na stacjach wyładowania, polowe składy MPS szczebla nadrzędnego, kompanie masowego tankowania mogące przewozić w cysternach 300t paliw, stacje paliwowe gospodarki narodowej i stacje paliwowe jednostek wojskowych znajdujących się w pobliżu dróg marszu.

4. METODY OKREŚLANIA ZUŻYCIA MPS W MARSZU

W dotychczasowej praktyce przyjmuje się dwie metody określania zużycia MPS:

- statystyczna;
- matematyczna.

4.1. Metoda statystyczna

Metoda statystyczna polega na wykorzystaniu doświadczeń minionych i współczesnych wojen, konfliktów lokalnych i prowadzonych ćwiczeń z wojskami, według których przyjmuje się średnie zużycie na 100km marszu bez uwzględniania współczynników wpływających na wielkość tego zużycia.

Wielkość tego zużycia przedstawiono w tabeli 2.

4.2. Metoda matematyczna

Metoda matematyczna polega na wyznaczeniu zużycia MPS przy wykorzystaniu mikrokomputerów. Jest to metoda, w której uwzględnia się czynniki wpływające na wielkość zużycia MPS, do których należy zaliczyć:

- jednostkowe zużycie paliwa na 100km w poszczególnych rodzajach pojazdów;
- zasięg poszczególnych rodzajów pojazdów na jednostce napełnienia;

- długość drogi marszu;
- uaktualnioną masę jednostki napędzenia MPS.

Dla określenia zużycia paliw dla poszczególnych rodzajów pojazdów przyjęto następujący wzór:

$$Z = \frac{L * (1 + k_{mt}) * (1+k_e)}{S}$$

gdzie:

- Z - zużycie poszczególnych rodzajów paliw wyrażone w jn;
- L - długość drogi marszu;
- k_{mt} - współczynnik manewrowo-taktyczny;
- k_e - współczynnik eksploatacyjny;
- S - zasięg danego rodzaju pojazdów na 1jn według zasadniczej należności.

Wielkość zużycia MPS obliczona metodą matematyczną wynika przede wszystkim z wartości przyjmowanych współczynników k_{mt} i k_e .

4.2.1. Współczynnik eksploatacyjny "ke"

Współczynnik eksploatacyjny " k_e " określa zwiększenie zużycia MPS w zależności od warunków terenowych i atmosferycznych odbiegających od warunków w jakich została określona zasadnicza norma zużycia MPS $(1+k_e)^{*/}$.

**/ Jedyńka we wzorze oznacza zasadniczą normę zużycia MPS, a " k_e " zwiększenie normy zużycia wskutek odmiennych warunków pracy silnika jakie ustala ZNZ.*

Współczynnik " k_e " na marsz może przyjmować następujące wartości

- dla:
- samochody 0.2 - 0.4;
 - transportery opancerzone kołowe 0.1 - 0.3;
 - transportery opancerzone gąsienicowe 0.1 - 0.2.

4.2.2. Współczynnik manewrowy " k_{mt} "

Współczynnik " k_{mt} " określa zwiększone zużycie MPS w wyniku manewrowego charakteru prowadzenia marszu wojsk własnych i intensywności oddziaływania nieprzyjaciela na wojska i teren.

Wielkość współczynnika " k_{mt} " jest uzależniona od tempa marszu. Im tempo marszu większe tym " k_{mt} " przyjmujemy mniejszy. Można stwierdzić, że wielkość współczynnika " k_{mt} " jest odwrotnie proporcjonalna do szybkości marszu, a wprost proporcjonalna do oddziaływania nieprzyjaciela na wojska i teren.

W zależności od przewidywanej sytuacji bojowej współczynnik " k_{mt} " może przyjąć wartości dla marszu 0.2 - 0.8. Współczynnik ten jest równy dla wszystkich typów pojazdów w marszu. Wynika to z faktu, że rejon docelowy jest jeden dla wszystkich typów pojazdów mechanicznych i wszystkie pojazdy muszą pokonać tę samą drogę.

W praktyce szkoleniowej wojsk przyjmuje się dla marszu jednolite wartości współczynników, które wynoszą:

Współczynnik Rodzaj pojazdu	k_{mt}	k_e
Pojazdy kołowe	0.3	0.4
Pojazdy gąsienicowe	0.3	0.35

Stosowanie w praktyce dwóch różnych współczynników jest pewnym utrudnieniem. W związku z tym sprowadzono je do jednego współczynnika syntetycznego oznaczając go jako "K".

Współczynnik syntetyczny "K" wynika z iloczynu współczynnika manewrowo-taktycznego (k_{mt}) i współczynnika eksploatacyjnego (k_e).

Wzór na wyznaczenie zużycia paliw dla poszczególnych rodzajów pojazdów mechanicznych przyjmuje postać:

$$Z = \frac{L * K}{S}$$

Wartości liczbowe współczynnika "K" są następujące dla ogółu wojsk wykonujących marsz:

- pojazdy kołowe 1.82;
- pojazdy gąsienicowe 1.75.

4.3. Obliczanie zużycia MPS przez poszczególne kolumny

Zużycie materiałów pędnych i smarów przez poszczególne kolumny marszowe będzie obliczane wg następującego wzoru:

$$Z_u = K * \frac{L * j_n}{100}$$

gdzie:

Z_u - zużycie MPS w tonach;

K - współczynnik zużycia MPS;

L - wielkość przebytej drogi marszu;

j_n - wielkość jednostki napełnienia kolumny marszowej
w tonach.

5. OPIS PROGRAMU "ZAOPATRZENIE W MPS BRYGADY ZMECHANIZOWANEJ W MARSZU" (MPSMARSZ)

5.1. Przeznaczenie programu

Program **MPSMARSZ** jest programem komputerowym umożliwiającym wykonanie potrzebnych obliczeń i kalkulacji przy organizacji zabezpieczenia marszu brygady zmechanizowanej w materiały pędne i smary.

5.1.1. Język programowania

Program **MPSMARSZ** został poracowany i oprogramowany w języku **TURBO PASCAL** wersja 6.0. uruchomienie i testowanie programu odbyło się pod kontrolą edytora, kompilatora i debagera firmy **BORLAND INTERNATIONAL**.

5.1.2. Sprzęt

Konfiguracja sprzętowa na której opracowano program oparta była o mikrokomputer klasy **IBM PC 386/486** i przedstawiała się następująco:

MIKROKOMPUTER

Procesor	80486/80386;
Zegar	33 MHz;
Pamięć	8 Mb;

Karta graficzna	SVGA color 1Mb;
HD	340Mb (12ms);
Fllopy	3.5' 1.44Mb, 5.25' 1.2 Mb;
Porty	2 szeregowo, 1 rOwnoległy;
Monitor	SVGA color;
DOS	5.0.

DRUKARKA

Drukarka Epson FX 1050.

Do poprawnego funkcjonowania programu wystarczająca będzie konfiguracja oparta o mikrokomputer klasy IBM PC .

5.1.3. Pliki systemu

W skład oprogramowania, oprócz programu MPSMARSZ wchodzi następujące pliki:

- * mpsmarsz.exe
- * egapl.exe
- * kybpl.exe
- * printpl.exe.

5.2. Uruchomienie programu

Aby program MPSMARSZ poprawnie funkcjonował muszą być spełnione następujące warunki:

- a) Na nośniku (dysku, dyskietce) gdzie znajduje się program muszą znajdować się następujące zbiory systemowe:

- * cysterny;
- * jedn_nap;
- * kolumny;
- * kolumnym.

Jeżeli powyższy warunek zostanie spełniony wystarczy wykonać jedną z następujących czynności:

- a) wprowadzić z klawiatury nazwę programu: MPSMARSZ.EXE;
- b) naprowadzić kursor na nazwę programu: MPSMARSZ.EXE
i nacisnąć klawisz <ENTER>.

Po wykonaniu tych czynności program rozpocznie działanie.

5.3 Obsługa programu

Program MPSMARSZ rozpoczyna swoje działanie od wyprowadzenia na ekran mikrokomputera nazwy programu. Na dole ekranu pojawia się polecenie, które musi wykonać użytkownik - należy nacisnąć klawisz <ENTER>.

Wygląd pierwszego ekranu jest następujący:

CENTRUM INFORMATYKI
KATEDRA LOGISTYKI WYDZIAŁU WOJSK LĄDOWYCH

prezentuje program:

ZAOPATRYWANIE W MPS BRYGADY ZMECHANIZOWANEJ
WYKONUJĄCEJ MARSZ

opracowany przez:

płk dr inż. Zbigniew KLIMKIEWICZ

płk dr Aleksander ŁOJKO

Naciśnij klawisz - ENTER

Po naciśnięciu klawisza <ENTER> na ekranie pojawi się menu główne programu w następującej postaci:

PROGRAM ANALIZY ZUŻYCIA I SPOSOBU PRZYDZIAŁU MPS

Główne menu

1. Zestawienie masy jednostek napełnienia pododdziałów
2. Sformowanie kolumny marszowej
3. Obliczanie jednostek napełnienia poszczególnych kolumn
4. Obliczanie zużycia mps przez poszczególne kolumny
5. Podział posiadanych sił i środków służby mps
6. Operacje dyskowe
7. Zakończenie pracy z programem

Polecenie - _

Użytkownik wybiera funkcję, którą ma realizować program, jej numer wprowadza do komputera i naciska klawisz <ENTER>.

5.3.1. Zestawienie masy jednostek napełnienia pododdziałów

Program komputerowy MPSMARSZ ma "wszyste" pełne zestawienie jednostek napełnienia pododdziałów brygady zmechanizowanej na podstawie etatu ćwiczebnego.

Istnieje możliwość dokonania modyfikacji tego zestawienia przez użytkownika oraz wykorzystanie programu w innych jednostkach rodzajów wojsk.

Po wyprowadzeniu na ekran monitora ZESTAWIENIA MASY JEDNOSTEK NAPEŁNIENIA PODODDZIAŁÓW - na dole ekranu pojawia się pytanie:

"Czy chcesz dokonać zmian (t/n)?"

W przypadku, gdy użytkownik chce dokonać zmian wprowadza znak <t> i naciska <ENTER>. Wówczas komputer pyta, w którym pododdziale należy dokonać tych zmian a następnie ukazują się nazwy paliw i użytkownik wpisuje nowe wartości. Aby dane te można było zachować na stałe wykorzystujemy opcję: OPERACJE DYSKOWE - pkt 1: Zapis tabeli jednostek napełnienia.

Przykładowe ZESTAWIENIE MASY JEDNOSTEK NAPEŁNIENIA PODODDZIAŁÓW przedstawiono poniżej:

ZESTAWIENIE MASY JEDNOSTEK NAPEŁNIENIA PODODDZIAŁÓW BRYGADY

Oddział	BS	ONPK	ONPG	suma ON
bcz/T-72/	1.27	0.80	34.02	34.82
bcz/T-55/	1.27	0.80	27.30	28.10
bz	2.46	1.52	0.80	2.32
bpzmot	2.46	2.10	0.46	2.56
dah	2.60	1.36	6.67	8.03
daplot	2.05	2.86	2.05	4.91
dappanc	1.81	----	----	----
bdow	2.46	3.10	1.10	4.20
kr	1.30	----	----	-----
ksap	1.04	1.39	4.22	5.61
kmed	1.04	----	----	----
kzaop	8.04	7.56	----	7.56
krem	8.35	1.66	2.66	4.32

Czy chcesz zmienić jn pododdziału (t/n) ?

Istnieje możliwość wyprowadzenia informacji przedstawionej na ekranie monitora na drukarkę. W tym celu należy jednocześnie nacisnąć klawisze <Shift> i <Print Screen>.

5.3.2. Sformowanie kolumny marszowej.

Po wybraniu tej opcji z menu głównego programu na ekranie monitora pojawia się tzw. menu pomocnicze w postaci:

F O R M O W A N I E K O L U M N

1. Przeglądanie istniejących kolumn
2. Sformowanie nowej kolumny
3. Powrót do głównego menu

Polecenie.....

Użytkownik w zależności od aktualnych potrzeb wybiera żadaną funkcję poprzez naciśnięcie jednego z wybranych klawiszy <1>, <2>, <3>, oraz klawisza <ENTER>.

5.3.2.1. Sformowanie nowej kolumny

Po wybraniu przez użytkownika z menu FORMOWANIE KOLUMN opcji 2 na ekranie monitora pojawi się komunikat:

Podaj numer kolumny:

Użytkownik wprowadza numer tworzonej kolumny marszowej i naciska klawisz <ENTER>. Użytkownik podaje numer tworzonej kolumny - liczbę z przedziału [1...10]. Wówczas na ekranie pojawia się następujący komunikat:

DOPUSZCZALNE NAZWY PODODDZIAŁÓW

bcz/T-72/	bcz/T-55/	bz	bpzmot	dah	daplot	bdow
dappanc	kr	ksap	kmed	kzaop	krem	

Kolumna nr: 1

Wprowadź kolejną nazwę pododdziału:

Użytkownik wprowadza nazwę pododdziału ze zbioru DOPUSZCZALNYCH NAZW PODODDZIAŁÓW, który wchodzi w skład aktualnie tworzonej kolumny. Pododdziały mogą być dzielone np:

1/2 bcz/T-72/, 1/2 bpzmot, 1/6 ksap itp.

Po wprowadzeniu nazwy pododdziału i naciśnięciu klawisza <ENTER> na ekranie monitora pojawi się komunikat-polecenie:

Podaj procent ukończenia: ...

Użytkownik podaje procent ukończenia pododdziału, wprowadzonego do składu formowanej kolumny (jest to liczba z przedziału [0 - 100]) i naciska klawisz <ENTER>.

W dolnej części ekranu pojawia się nazwa wprowadzonego do składu formowanej kolumny pododdziału, a program pyta się o nazwę kolejnego pododdziału wchodzącego w skład formowanej kolumny.

Postać ekranu może być następująca:

DOPUSZCZALNE NAZWY PODODDZIAŁÓW

bcz/T-72/	bcz/T-55/	' bz	bpzmot	bdow	dah	daplot
dappanc	kr	ksap	kmed	kzaop	krem	

Kolumna nr: 1

Wprowadź kolejną nazwę pododdziału:

Podaj procent ukończenia:

1/2 bcz/T-72/

1/2 bpzmot

1/6 ksap

1/2 das

1/9 krem

1/2 kzaop

Gdy wprowadzimy ostatni pododdział do składu formowanej kolumny, to naciśnięcie klawisza <ENTER> powoduje zakończenie formowania kolumny i powrót do menu pomocniczego FORMOWANIE KOLUMN.

5.3.2.2. Przeglądanie istniejących kolumn

Po wybraniu przez użytkownika z menu pomocniczego tej opcji spowoduje, że program na ekran monitora wyprowadzi następujący komunikat:

Podaj numer kolumny: ...

Użytkownik wyprowadza numer kolumny, której skład chce uzyskać na ekranie monitora.

Numer kolumny - liczba z przedziału [1..ilość_kol] (gdzie ilość_kol - liczba utworzonych kolumn, maksymalnie może mieć ona wartość 10).

Po naciśnięciu klawisza <ENTER> na ekranie monitora pojawi się skład organizacyjny żądanej kolumny.

Przykład:

SKŁAD KOLUMNY NR 1

1/2 bcz/T-72/

1/2 bpzmot

1/6 ksap

1/2 das

1/9 krem

1/2 kzaop

Naciśnij ENTER

Po naciśnięciu klawisza <ENTER> następuje powrót do menu pomocniczego - FORMOWANIE KOLUMN.

Możemy przeglądać kolejne kolumny lub po wybraniu opcji nr 3 - Powrót do menu głównego i naciśnięciu klawisza <ENTER> następuje powrót do menu głównego.

5.3.3. Obliczanie masy jednostek napełnienia kolumn marszowych.

Komputer oblicza i wyprowadza na ekran monitora masy jednostek napełnienia sformowanych kolumn ugrupowania marszowego brygady. Zestawienie ma następującą postać:

ZESTAWIENIE MASY JEDNOSTEK NAPEŁNIENIA KOLUMN

kolumny	BS	ONPK	ONPG	Suma ON
1	6.33	4.98	4.33	9.31
2	6.69	5.66	2.25	7.91

Naciśnij ENTER

Po naciśnięciu przez użytkownika klawisza <ENTER> następuje powrót do menu głównego programu.

5.3.4. Obliczanie zużycia MPS przez poszczególne kolumny

Wybierając z menu głównego tę opcję na ekranie monitora pojawia się następujący komunikat:

OBLICZANIE ZUŻYCIA MPS PRZEZ POSZCZEGÓLNE KOLUMNY

Zużycie będzie obliczane według następującego wzoru:

$$Zu = K \frac{L * jn}{100}$$

Oznaczenia:

Zu - zużycie mps w tonach

L - wielkość przebytej drogi w marszu, w I rzucie, w II rzucie

jn - wielkość jednostki napełnienia kolumny marszowej w tonach

K - współczynnik zużycia mps

Naciśnij <ENTER>

Po naciśnięciu klawisza <ENTER> na ekranie monitora pojawi się następujący komunikat:

PARAMETRY WEJŚCIOWE

Podaj wartości współczynników K dla poszczególnych rodzajów paliw

KBS = 0.28

KONPK = 0.28

KONPG = 0.30

M A R S Z

Podaj długość marszu do rejonu dwugodzinnego postoju L1 = 150

Podaj długość całego marszu L12 = 320

Naciśnij <ENTER>

Po naciśnięciu przez użytkownika klawisza <ENTER> na ekranie monitora pojawi się następujący komunikat:

OBLICZANIE ZUŻYCIA MPS W CZASIE MARSZU

1. Do rejonu dwugodzinnego postoju
2. Od rejonu dwugodzinnego postoju do rejonu dziennego (nocnego) odpoczynku
3. W czasie całego marszu
4. Zadanie innych parametrów marszu
5. Powrót do głównego menu

Polecenie - ...

Użytkownik z menu pomocniczego wybiera potrzebną opcję obliczeń, jej numer wprowadza do programu oraz naciska klawisz <ENTER>.

W zależności od wybranej opcji obliczeń z menu pomocniczego użytkownik otrzymuje następujące komunikaty na ekranie monitora:

ZUŻYCIE MPS NA DRODZE DO REJONU DWUGODZINNEGO POSTOJU

Kolumna	ZuBS	ZuONPK	ZuONPG	sumaON	
1	2.66	2.09	1.95	13.09	KBS = 0.28
2	3.16	2.63	2.36	9.05	KONPK = 0.28
-----					KONPG = 0.30
suma	5.82	4.73	4.31	22.14	

L1 = 150.0 km
L2 = 170.0 km
L12 = 320.0 km

Naciśnij <ENTER>

ZUZYCIE MPS NA DRODZE OD REJONU POSTOJU DO DO REJONU DZIENNEGO ODPOCZYWK

Kolumna	ZuBS	ZuONPK	ZuONPG	sumaON	
1	3.01	2.37	2.21	10.26	KBS = 0.28
2	3.58	2.98	2.68	10.26	KONPK = 0.28
-----					KONPG = 0.30
suma	6.59	5.36	9.20	20.52	

L1 = 150.0 km
L2 = 170.0 km
L12= 320.0 km

Naciśnij <ENTER>

ZUZYCIE MPS W CZASIE CAŁEGO MARSZU

Kolumna	ZuBS	ZuONPK	ZuONPG	sumaON	
1	5.67	4.46	4.16	19.31	KBS = 0.28
2	6.74	5.62	5.04	19.31	KONPK = 0.28
-----					KONPG = 0.30
suma	12.41	10.08	18.41	38.33	

L1 = 150.0 km
L2 = 170.0 km
L12= 320.0 km

Naciśnij <ENTER>

W każdym z przedstawionych komunikatów po naciśnięciu klawisza <ENTER> program powraca do menu pomocniczego.

Powrót do menu głównego następuje po wybraniu opcji "5" menu pomocniczego.

5.3.5. Podział posiadanych sił i środków służby MPS

Po wybraniu tej opcji obliczeń z menu głównego na ekranie monitora pojawia się następujące menu pomocnicze:

ANALIZA I PODZIAŁ POSIADANYCH SIŁ I ŚRODKÓW SŁUŻBY MPS

1. Wprowadzanie danych o siłach i środkach służby mps
2. Wyświetlanie danych
3. Podział sił i środków służby mps
4. Powrót do głównego menu

Polecenie - ...

Użytkownik wybiera numer potrzebnej opcji i wprowadza go do programu oraz naciska klawisz <ENTER>.

Realizacja poszczególnych opcji z menu pomocniczego przebiega w sposób następujący:

5.3.5.1. Wprowadzanie danych o siłach i środkach służby MPS

Ta opcja programu umożliwia użytkownikowi wprowadzenie danych o posiadanych siłach i środkach służby MPS.

Na ekranie pojawiają się poszczególne rodzaje środków transportowych służby MPS, a użytkownik podaje ich ilość w odniesieniu do benzyny samochodowej (BS) i oleju napędowego (ON).

W dolnej części ekranu podawana jest automatycznie sumaryczna

ilość posiadanych środków MPS - benzyny samochodowej - ilość BS oraz oleju napędowego - ilość ON.

Po wprowadzeniu wszystkich danych o posiadanych siłach i środkach służby MPS ekran monitora może wyglądać następująco:

WPROWADZENIE DANYCH O SIŁACH I ŚRODKACH SŁUŻBY MPS

środek transportu	ilość sztuk	
	BS	ON
samochód transportowy z beczkami STRB	0	2
cysterna dystrybutor 4.5 CD4.5	1	3
cysterna dystrybutor 5 CD5W	0	0
cyst. dyst. paliwowo-olejowy CDPO4W	0	0
cysterna dystrybutor 7.5 CD7.5	0	1
samochód transportowy z karnistr.STRK	0	0
cysterna dystrybutor CD5WK	0	0
cysterna przyczepa CP	1	1
przyczepa transportowa z beczkami PT	1	0
cysterna paliwowa-dystrybutor CP11	0	1

ilość BS 8.2 t

ilość ON 25.6 t

5.3.5.2. Wyświetlanie danych

Wykorzystując tę opcję menu pomocniczego użytkownik może uzyskać na ekranie monitora informacje o posiadanych siłach i środkach służby MPS.

Postać otrzymanych danych jest następująca:

SILY I ŚRODKI SŁUŻBY MPS

środek transportu	ilość:	
	BS	ON
samochód transportowy z beczkami STRB	---	2
cysterna dystrybutor 4.5 CD4.5	1	3
cysterna dystrybutor 5 CD5W	---	---
cyst. dyst. paliwowo-olejowy CDPO4W	---	---
cysterna dystrybutor 7.5 CD7.5	---	1
samochód transportowy z karnistr.STRK	---	---
cysterna dystrybutor CD5WK	---	---
cysterna przyczepa CP	1	1
przyczepa transportowa z beczkami PT	1	---
cysterna paliwowa-dystrybutor CP11	---	1

Naciśnij <ENTER>

Po naciśnięciu przez użytkownika klawisza <ENTER> na ekranie monitora pojawi się następujący komunikat:

PORÓWNANIE POTRZEB Z MOŻLIWOŚCIAMI

Zużycie		Przydział	
BS	ON	BS	ON
12.41 t	57.90 t	8.19 t	40.75 t

POSIADANA ILOŚĆ PALIW STANOWI

BS	ON
66.0 % zużycia	70.4 % zużycia

Naciśnij <ENTER>

Po naciśnięciu klawisza <ENTER> program powraca do menu pomocniczego.

Gdy chcemy wrócić do menu głównego programu należy wybrać opcję "4" z "menu pomocniczego - Powrót do menu głównego i nacisnąć klawisz <ENTER>.

5.3.5.3. Podział sił i środków służby MPS

Po wybraniu tej opcji z menu pomocniczego na ekranie monitora pojawia się komunikat - polecenie:

Podaj priorytet przydziału paliwa dla czołgów (1-3)

Autorzy programu uważali za celowe wprowadzenie takiej modyfikacji, ponieważ czołgi z jednej strony stanowią największy potencjał bojowy brygady, z drugiej zaś mają najmniejszy zasięg na jednej jednostce napełnienia.

Użytkownik może skompensować ten dysonans wprowadzając priorytet dla czołgów np. 1.5 lub 1.2 itp.

Transport z MPS przydzielony jest do kolumn w wielkości proporcjonalnej do zużycia. Algorytm przydziału transportu MPS ma następującą postać:

Algorytm przydziału transportu MPS do poszczególnych kolumn

↓

Ze wszystkich kolumn marszowych
wybiera się kolumnę o największym
zużyciu paliwa

↓

Przydzielenie określonego ~~środku~~
(środków) transportowych z MPS

↓

Przydzielone środki odejmij od
ogólnej liczby środków dystrybucyjnych MPS

↓

Sprawdź warunek
Czy zostały obsłużone wszystkie kolumny?

↓ N

Z grupy nie obsłużonych kolumn wybierz
kolumnę o największym zużyciu paliwa

↓ T

Zestawienie "PODZIAŁ SIŁ I ŚRODKÓW SŁUŻBY MPS"

Przy dzieleniu środków transportowo-dystrybucyjnych do poszczególnych kolumn marszowych obowiązują określone priorytety.

Samochody z beczkami oraz cysterny-dystrybutory CDPO4W przydzielane są w pierwszej kolejności do kolumn, w których znajdują się czołgi (sprzęt ten najlepiej nadaje się do uzupełniania paliwa w czołgach).

Samochody z karnistrami w pierwszej kolejności przydzielane są do kolumn, w których znajduje się SD i CKZL (w kolumnach tych będzie najwięcej samochodów).

Przykładowe zestawienie "PODZIAŁU SIŁ I ŚRODKÓW SŁUŻBY MPS" przedstawiono poniżej:

PODZIAŁU SIŁ I ŚRODKÓW SŁUŻBY MPS

	STRB	CD4.5	CD5W	CDPO4W	CD7.5	STRK	SD5WK	CP	PT	CD7.5+
kol.	bs on	bs on	bs on	bs on	bs on	bs on	bs on	bs on	bs ob	bs
1	-- 1	-- 1	-- --	-- --	-- --	-- --	-- --	1 --	1 --	--
2	-- 1	1 2	-- --	-- --	-- 1	-- --	-- --	-- 1	-- --	--
3	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	--
4	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	--
5	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	--
6	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	--
7	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	--
8	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	--
9	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	--
10	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	-- --	--

Naciśnij <ENTER>

Po naciśnięciu klawisza <ENTER> otrzymujemy na ekranie monitora następujący komunikat:

BILANS ZUZYCIA PALIW

	BS	ON
W rejonie wyjściowym pozostało:	14.68	37.81
DO PEŁNYCH ZAPASÓW RUCHOMYCH:		
Przy sprzęcie należy uzupełnić:	6.20	18.77
W brygadzie należy uzupełnić:	11.02	55.71

Naciśnij <ENTER>

Po naciśnięciu klawisza <ENTER> program powraca do menu pomocniczego.

5.3.6. Operacje dyskowe

Wybór tej opcji z menu głównego programu powoduje, że na ekranie monitora pojawia się tzw. menu pomocnicze w postaci:

OPERACJE DYSKOWE

1. Zapis tabeli jednostek napełnienia
2. Odczyt tabeli jednostek napełnienia
3. Zapis składu kolumn
4. Odczyt składu kolumn
5. Zapis środków służby MPS
6. Odczyt środków służby MPS
7. Powrót do głównego menu

Polecenie - ...

W zależności od potrzeb użytkownik wybiera jedną z przedstawionych opcji i naciska klawisz <ENTER>.

Wybór opcji o numerach 1, 3, 5 spowoduje zapis odpowiedniego zbioru na dysk twardy lub dyskietkę.

5.3.6.1. Odczyt tabeli jednostek napełnienia

Po wybraniu tej opcji z menu pomocniczego - OPERACJE DYSKOWE, użytkownik na ekranie monitora otrzymuje następujące zestawienie:

TABELA JEDNOSTEK NAPEŁNIENIA

Nr pododdz	BS	ONPK	ONPG
bcz/T-72/	1.27	0.80	34.02
bcz/T-55/	1.27	0.80	27.30
bz	2.46	1.52	0.80
bpzmot	2.46	2.10	0.46
dah	2.60	1.36	6.67
daplot	2.05	2.86	2.05
dappanc	1.81	----	----
bdow	2.46	3.10	1.10
kr	1.30	----	----
ksap	1.04	1.39	4.22
kmed	1.04	----	----
kzaop	8.04	7.56	----
krem	8.35	1.66	2.66

Naciśnij <ENTER>

Po naciśnięciu klawisza <ENTER> na ekranie monitora pojawia się menu pomocnicze - OPERACJE DYSKOWE.

W przypadku, gdy zbiór o nazwie JEDN_NAP jest zbiorem pustym, na ekranie monitora pojawi się następujący komunikat:

Brak zbioru JED_NAP

Naciśnij <ENTER>

Po naciśnięciu klawisza <ENTER> na ekranie monitora pojawia się menu pomocnicze - OPERACJE DYSKOWE.

5.3.6.2. Odczyt składu kolumn

Wybierając tę opcję z menu pomocniczego - OPERACJE DYSKOWE użytkownik otrzymuje informacje o składzie kolumn zapisanych uprzednio do zbioru "KOLUMNY" na dysku (dyskiecie) w następującej postaci:

SKŁAD KOLUMNY NR 1

bpz	1/2da	1/3daplot
1/6ksap	1/2kzaop	

Naciśnij <ENTER>

Po naciśnięciu klawisza <ENTER> uzyskujemy na ekranie monitora informacje o składzie kolejnej kolumny.

SKŁAD KOLUMNY NR 2

1/2bcz/T-72/

1/2bpzmot

1/2das

1/6krem

1/2kzaop

Naciśnij <ENTER>

Po przedstawieniu informacji o składzie wszystkich kolumn zapisanych w zbiorze "KOLUMNYM" program wyprowadza na ekran monitora menu pomocnicze - OPERACJE DYSKOWE i czeka na decyzje użytkownika.

W przypadku, gdy zbiór o nazwie KOLUMNYM jest zbiorem pustym, na ekranie monitora pojawi się następujący komunikat:

Brak zbioru KOLUMNYM

Naciśnij <ENTER>

Po naciśnięciu klawisza <ENTER> program wyprowadza na ekran monitora menu pomocnicze - OPERACJE DYSKOWE i czeka na decyzje użytkownika.

Wybierając opcję "Powrót do głównego menu" - 7, program wraca do menu głównego programu.

5.3.6.3. Odczyt środków służby MPS

Wybierając tę opcję z menu pomocniczego - "OPERACJE DYSKOWE" użytkownik otrzymuje informacje o stanie środków służby MPS, które są zapisane na dysku w zbiorze o nazwie CYSTERNY. Postać otrzymanych informacji jest następująca:

SILY I SRODKI SLUZBY MPS

srodek transportu		ilosc sztuk	
		BS	ON
samochod transportowy z beczkami STRB		1	1
cysterna dystrybutor 4.5	CD4.5	1	2
cysterna dystrybutor 5	CD5W	---	---
cyst. dyst. paliwowo-olejowy	CDPO4W	---	---
cysterna dystrybutor 7.5	CD7.5	---	1
samochod transportowy z karnistr.STRK		---	---
cysterna dystrybutor	CD5WK	---	---
cysterna przyczepa	CP	1	1
przyczepa transportowa z beczkami	PT	---	2
cysterna paliwowa dystrybutor	CP11	1	2

Nacisnij <ENTER>

Po naciśnięciu klawisza <ENTER> program powraca do menu pomocniczego.

W sytuacji, gdy na dysku jest brak zbioru o nazwie "CYSTERNY", na ekran monitora zostanie wyprowadzony komunikat:

Brak zbioru CYSTERNY

Nacisnij <ENTER>

Po naciśnięciu klawisza <ENTER> następuje powrót do menu pomocniczego - OPERACJE DYSKOWE i program czeka na podjęcie decyzji przez użytkownika.

5.3.7. Zakończenie pracy programu

Wybór tej opcji z głównego menu programu powoduje, że program kończy pracę i następuje powrót do systemu operacyjnego DOS.

Wykonano w 2 egz.

Egz. 1,2 - Biblioteka Zbiorów Niejawnych

Wyk. płk Klimkiewicz Z.

Druk. E.J.

Nr ks. RWD 32/pf 3/ pf 37

Załącznik

Program źródłowy MPSMARSZ

```
2 Program epsmarsz;
3 uses Crt, DOS, printer;
4 Const
5     ilosc_podod = 13;
6     ilosc_kol  = 10;
7     ilosc_sr   = 10;
8     benzyna   = 1;
9     olej      = 2;
10    STR8      = 1;
11    CD45      = 2;
12    CDSW      = 3;
13    CDF04W    = 4;
14    CD75      = 5;
15    STRK      = 6;
16    CDSWK     = 7;
17    CP        = 8;
18    PT        = 9;
19    CD75CP11  = 10;

21 Type
22     jedn_nap = record
23         BS : Real;
24         ONPK : Real;
25         ONPG : Real;
26         ONCZ : Real;
27     end;
28     opis_kol = record
29         bs,on : byte;
30     end;
31     opis_zap = record
32         ben : real;
33         pbs : real;
34         oncz : real;
35         poncz : real;
36         onkg : real;
37         ponkg : real;
38     end;
39     opis_przydz = record
40         ben : integer;
41         on : integer;
42     end;
43     skl_kol = record
44         nr_odd : Integer;
45         wk_odd : string[3];
46         ulamek : Real;
47         proc : Real;
48         strt : Real;
49     end;
50     opis_cyst = record
51         naz : String[40];
52         ben : Integer;
53         on : Integer;
54         zbn : Integer;
```

```
55         zon      : Integer;
56         pojem   : Real;
57     end;
58     Nazwa = String[14];

60 var
61     wyj : text;
62     kolejnosc : array[1..ilosc_kol] of opis_kol;
63     przydzial_BS, przydzial_ON : real;
64     tablica_przydzialu : Array[1..ilosc_kol, 1..ilosc_sr] of opis_przydz;
65     zap_kol      : Array[1..ilosc_kol] of opis_zap;
66     cysterny     : Array[1..ilosc_sr] of opis_cyst;
67     tab_jed_nap : Array[1..ilosc_podod] of jedn_nap;
68     nazwa_podod : Array[1..ilosc_podod] of Nazwa;
69     kolumny     : Array[1..ilosc_kol, 1..ilosc_podod] of skl_kol;
70     polecenie_da : Char;
71     k, i, numer_odd, rejs : Integer;
72     nazwa_odd      : Nazwa;
73     numer_kol     : Integer;
74     kbd           : char;
75     numer         : String[4];
76     kreski        : Byte;
77     wiersz        : String[70];
78     nie_wym      : Boolean;
79     wykonano     : Boolean;
80     MBS, MON, MONPK, MONPG, ul : Real;
81     KBS, KONPK, KONPG : Real;
82     ZuBS, ZuONPK, ZuONPG, ZuON, suma : Real;
83     cyfra_1, cyfra_2, nr_linii : Integer;
84     suma_ben, suma_on : Real;
85     ilosc_ben, ilosc_on : Integer;
86     pomijac : Boolean;
87     L1, L12, L2 : Real;
88     sa_cysterny : Boolean;
89     przydzielono, czolgi, jest : Boolean;
90     paliwo, cyst : integer;
91     BufLen : integer;
92     priorytet : real;
93     nr_kla : integer;

96 function mamy(cyster : integer) : Boolean;
97     begin
98         mamy:=false;
99         with cysterny[cyster] do
100             begin
101                 case paliwo of
102                     benzyna : if ((ben-zbn)>0) then mamy:=true;
103                     olej : if ((on -zon)>0) then mamy:=true;
104                 end;
105             end;
106         end;

108 function potrzebuja_czolgi : boolean;
```

```
109 begin
110   potrzebuja_czolgi:=false;
111   for k:=1 to ilosc_kol do
112     begin
113       with zap_kol[k] do
114         begin
115           if ((onkg-poncz) > 0) then
116             begin
117               potrzebuja_czolgi:=true;
118               exit;
119             end;
120         end;
121     end;
122 end;

124 Procedure szuk_max_czolgi;
125 var
126   zap_on : real;
127 begin
128   zap_on:=0;
129   jest:=false;
130   for k:=1 to ilosc_kol do
131     begin
132       with zap_kol[k] do
133         begin
134           if ((onkg-poncz) > zap_on) then
135             begin
136               zap_on:=onkg-poncz;
137               jest:=true;
138               nr_kla:=k;
139             end;
140         end;
141     end;
142 end;

144 procedure szuk_kol(n_podod : integer);
145 begin
146   nr_kla:=0;
147   for k:=1 to ilosc_kol do
148     begin
149       if kolumny[k,1].nr_odd<>0 then
150         begin
151           for j:=1 to ilosc_podod do
152             begin
153               if kolumny[k,j].nr_odd=n_podod then
154                 begin
155                   nr_kla:=k;
156                   exit;
157                 end;
158             end;
159           end;
160         end;
161     end;
```

```
163 Procedure szuk_max_zap(wlk_zap : integer);
164   var
165     zap_on,zapotrz : real;
166   begin
167     jest:=false;
168     zapotrz:=wlk_zap;
169     nr_klm:=0;
170     for k:=1 to ilosc_kol do
171       begin
172         with zap_kol[k] do
173           begin
174             case paliwo of
175               benzyna : begin
176                 if ((ben-pbs)>zapotrz) then
177                   begin
178                     jest:=true;
179                     nr_klm:=k;
180                     zapotrz:=ben-pbs;
181                   end;
182                 end;
183               olej   : begin
184                 if (onkg-ponkg)>0 then
185                   begin
186                     zap_on:=(onkg-poncz)*priorytet;
187                     if (zap_on>zapotrz) then
188                       begin
189                         zapotrz:=zap_on;
190                         nr_klm:=k;
191                         czolgi:=true;
192                         jest:=true;
193                       end;
194                     zap_on:=onkg-ponkg;
195                     if (zap_on>zapotrz) then
196                       begin
197                         zapotrz:=zap_on;
198                         nr_klm:=k;
199                         czolgi:=false;
200                         jest:=true;
201                       end;
202                     end;
203                   end;
204                 end;
205             end;
206           end;
207         end;
208       end;
209 Procedure przydzial(kol_num,cyst_rodz : integer);
210   begin
211     with tablica_przydzialu[kol_num,cyst_rodz] do
212       begin
213         case paliwo of
214           benzyna : ben:=ben+1;
215           olej   : on :=on +1;
216         end;
217       end;
218     end;
```

```
217     end;
218     with zap_kol[kol_num] do
219     begin
220         if (paliwo=olej) then
221         begin
222             przydzial_ON:=przydzial_ON+cysterny[cyst_rodz].pojem#0.82;
223             if czolgi then
224                 poncz:=poncz+cysterny[cyst_rodz].pojem#0.82
225             else
226                 ponkg:=ponkg+cysterny[cyst_rodz].pojem#0.82;
227         end
228     else
229     begin
230         pbs:=pbs+cysterny[cyst_rodz].pojem#0.78;
231         przydzial_BS:=przydzial_BS+cysterny[cyst_rodz].pojem#0.78;
232     end;
233     end;
234     with cysterny[cyst_rodz] do
235     begin
236         case paliwo of
237             benzyna : zbn:=zbn+1;
238             olej    : zon:=zon+1;
239         end;
240     end;
241 end;

243 Procedure przydz(kol_num,cyst_rodz: integer);
244 var
245     pojazdy : boolean;
246 begin
247     if ((cyst_rodz=STRB) or (cyst_rodz=CDPQ4W) or (cyst_rodz=PT)) and czolgi then
248         przydzial(kol_num,cyst_rodz)
249     else
250         if ((cyst_rodz=STRB) or (cyst_rodz=CDPQ4W) or (cyst_rodz=PT)) and not czolgi then
251         begin
252             if potrzebuja_czolgi then
253             begin
254                 szuk_max_czolgi;
255                 if jest then
256                 begin
257                     pojazdy:=czolgi;
258                     czolgi:=true;
259                     przydzial(kol_num,cyst_rodz);
260                     czolgi:=pojazdy;
261                 end;
262             end
263             else przydzial(kol_num,cyst_rodz);
264         end
265         else przydzial(kol_num,cyst_rodz);
266     end;

268 Procedure szuk_max_cyst;
269 var
270     wk_cyst, wk : real;
```

```
271 begin
272   jest:=false;
273   wlk_cyst:=0;
274   for k:=1 to ilosc_sr do
275     begin
276       with cysterny[k] do
277         begin
278           wlk:=0;
279           case paliwo of
280             benzyna : if ((ben-zbn)>0) then wlk:=pojem;
281             olej      : if ((on -zon)>0) then wlk:=pojem;
282           end;
283           if (wlc_cyst<wlc) then
284             begin
285               jest:=true;
286               wlk_cyst:=wlc;
287               cyst:=k;
288             end;
289           end;
290         end;
291     end;

293 Procedure rozdziel(cst : integer);
294 var
295   kolumn : integer;
296 begin
297   jest:=true;
298   while (mamy(cst) and jest) do
299     begin
300       przydzielono:=false;
301       szuk_max_zap(0);
302       if jest then
303         begin
304           kolumn:=nr_klm;
305           if (kolumn=nr_klm) then
306             begin
307               if mamy(STRK) then przydziel(nr_klm,STRK);
308               przydzielono:=true;
309             end
310           else
311             begin
312               if (kolumn=nr_klm) then
313                 begin
314                   if mamy(STRK) then przydziel(nr_klm,STRK);
315                   przydzielono:=true;
316                 end;
317             end;
318           if not przydzielono then
319             przydziel(kolumn,cst);
320         end;
321     end;
322 end;

324 Procedure rozdzielenie;
```

```
325 begin
326   rozdzial(CD75CP11);
327   rozdzial(CP);
328   rozdzial(CDP04W);
329   rozdzial(STRB);
330   rozdzial(CD5WK);
331   rozdzial(CD45);
332   rozdzial(CD5W);
333   rozdzial(CD75);
334   rozdzial(STRK);
335   rozdzial(PT);
336 end;

338 Procedure czekanie_na_return;
339 var
340   znak : Char;

342 begin
343   TextColor(15+128);
344   GoToxy(30,24);
345   Write('Nacisnij ENTER ');
346   repeat
347     znak:=readkey;
348   until znak = chr(13);
349 end;

351 Procedure wysw_rozdzialu;
352 begin
353   ClrScr;
354   TextColor(12);
355   gotoxy(1,1);
356   Writeln('          PODZIAW SIW I SRODKAW SLUBY MPS          ');
357   writeln;
358   TextColor(14);
359   Writeln('      STRB  CD4.5  CD5W  CDP04W  CD7.5  STRK  CD5WK  CP    PT  CD7.5+CP11');
360   writeln;
361   Writeln('kolum  bs on  bs on  bs on  bs on  bs on  bs on  bs on  bs on  bs on  bs on ');
362   writeln;
363   TextColor(15);
364   for k:=1 to ilosc_koi do
365     begin
366       write(' ',k,' ');
367       for j:=1 to ilosc_sr do
368         begin
369           if (tablica_przydzialu[k,j].ben=0) then write(' --')
370           else write(tablica_przydzialu[k,j].ben:4);
371           if (tablica_przydzialu[k,j].on =0) then write(' --')
372           else write(tablica_przydzialu[k,j].on:3);
373         end;
374       writeln;
375     end;
376 end;

378 Procedure ilosc_benzyny(n_r : integer);
```

```
379   var
380     l : integer;
381   begin
382     suma_ben:=0;
383     suma_on:=0;
384     for l:=1 to n_r do
385       begin
386         suma_ben:=suma_ben+cysterny[l].pojem#cysterny[l].ben#0.78;
387         suma_on :=suma_on +cysterny[l].pojem#cysterny[l].on #0.82;
388       end;
389     end;
```

```
391 Procedure obl_jed_nap_kol(kol : Integer);
392   var
393     akt_oddzial, nr_od : Integer;
394     ulam,pr, strat    : Real;
395   begin
396     akt_oddzial:=1;
397     HBS :=0;
398     HONPK:=0;
399     HONPG:=0;
400     if (kolumny[kol,1].nr_odd>0) then
401       begin
402         while (kolumny[kol,akt_oddzial].nr_odd > 0) do
403           begin
404             nr_od:=kolumny[kol,akt_oddzial].nr_odd;
405             ulam :=kolumny[kol,akt_oddzial].ulam;
406             pr :=kolumny[kol,akt_oddzial].proc/100.0;
407             with tab_jed_nap[nr_od] do
408               begin
409                 HBS :=HBS +BS #ulam#pr;
410                 HONPK:=HONPK+ONPK#ulam#pr;
411                 HONPG:=HONPG+ONPG#ulam#pr;
412               end;
413             akt_oddzial:=akt_oddzial+1;
414           end;
415         end;
416       end;
```

```
418 Procedure podzial;
419   var
420     jnBS,jnONPK,jnONPG : Real;
421     HBS,HON,HONPK,HONPG,UBS,UON,pHBS,pHON : Real;
422
423   begin
424     for k:=1 to ilosc_sr do
425       begin
426         cysterny[k].zbn:=0;
427         cysterny[k].zon:=0;
428       end;
429     if (L12 = 0) then
430       begin
431         TextColor(15);
432         gotoxy(20,18);
```

```
433     WriteLn('Nie podano parametrow marszu',chr(7));
434     czekanie_na_return;
435     wykonano:=false;
436 end
437 else
438     if not sa_cysterny then
439         begin
440             gotoxy(15,18);
441             WriteLn('Nie przydzielono srodkow sluzby aps',chr(7));
442             czekanie_na_return;
443             wykonano:=false;
444         end
445     else
446         begin
447             clrscr;
448             przydzial_BS:=0;
449             przydzial_ON:=0;
450             gotoxy(5,5);
451             write(' Podaj priorytet przydzialu paliwa dla czolgow (1-3) ');
452             gotoxy(58,5);
453             read(priorityet);
454             Clrscr;
455             for k:=1 to ilosc_kol do
456                 begin
457                     obl_jed_nap_kol(k);
458                     with zap_kol[k] do
459                         begin
460                             ben:=KBS*L12*MRS/100;
461                             pbs:=0;
462                             poncz:=0;
463                             onkg:=(KONPG*MONPG+KONPK*MONPK)*L12/100;
464                             ponkg:=0;
465                         end;
466                 end;
467             for k:=1 to ilosc_kol do
468                 for j:=1 to ilosc_sr do
469                     begin
470                         tablica_przydzialu[k,j].ben:=0;
471                         tablica_przydzialu[k,j].on :=0;
472                     end;
473             paliwo:=benzyna;
474             czolgi:=false;
475             if mamy(STRK) then
476                 begin
477                     if (nr_kla>0) then przydzial(nr_kla,STRK);
478                 end;
479             if mamy(STRK) then
480                 begin
481                     if (nr_kla>0) then przydzial(nr_kla,STRK);
482                 end;
483             paliwo:=olej;
484             if mamy(STRK) then
485                 begin
486                     if (nr_kla>0) then przydzial(nr_kla,STRK);
```

```
487         end;
488         if mamy(STRK) then
489             begin
490                 if (nr_klm>0) then przydzial(nr_klm,STRK);
491             end;
492         end;
493         czolgi:=false;
494         paliwo:=benzyna;
495         repeat
496             szuk_max_zap(0);
497             if jest then szuk_max_cyst;
498             if jest then przydzial(nr_klm,cyst);
499         until not jest;
500         czolgi:=true;
501         paliwo:=olej;
502         repeat
503             if mamy(STRB) then szuk_max_czolg;
504             if jest then przydzial(nr_klm,STRB);
505         until not (jest and mamy(STRB));
506         repeat
507             jest:=false;
508             if mamy(PT) then
509                 szuk_max_czolg;
510             if jest then przydzial(nr_klm,PT);
511         until not(jest and mamy(PT));
512         repeat
513             jest:=false;
514             if mamy(CDPO4W) then szuk_max_czolg;
515             if jest then przydzial(nr_klm,CDPO4W);
516         until not (jest and mamy(CDPO4W));
517         paliwo:=olej;
518         repeat
519             szuk_max_zap(0);
520             if jest then
521                 szuk_max_cyst;
522             if jest then
523                 przydzial(nr_klm,cyst);
524         until not jest;
525         paliwo:=olej;
526         rozdzielanie;
527         paliwo:=benzyna;
528         rozdzielanie;
529         wysw_rozdzialu;
530         czekanie_na_return;

532         {*****}
533         {#}
534         {# Obliczanie calkowitej masy napelnienia pulku #}
535         {#}
536         {*****}

538         jnBS :=0;
539         jnONPK:=0;
540         jnONPG:=0;
```

```
541     for k:=1 to ilosc_kol do
542         begin
543             obl_jed_nap_kol(k);
544             jnBS :=jnBS +MBS;
545             jnONPK:=jnONPK+MONPK;
546             jnONPG:=jnONPG+MONPG;
547         end;

549     {#####}
550     {# #}
551     {# Obliczenie ilosci paliwa jaka pozostala przy sprzecie #}
552     {# #}
553     {#####}

555     MBS :=jnBS #1.3-jnBS #KBS #L12/100;
556     MONPK:=jnONPK#1.3-jnONPK#KONPK#L12/100;
557     MONPG:=jnONPG -jnONPG#KONPG#L12/100;

559     {#####}
560     {# #}
561     {# Przejscie na rope i benzynie #}
562     {# #}
563     {#####}

565     pMBS:=MBS;
566     pMON :=MONPK+MONPG;
567     pMON:=MON;

569     {#####}
570     {# #}
571     {# W cysternach mamy suma_on i suma_ben #}
572     {# #}
573     {#####}

575     pMBS:=MBS+przydzial_BS;      { stan benzyny }
576     pMON:=MON+przydzial_ON;      { stan ropy }
577     ClrScr;
578     TextColor(12);
579     gotoxy(21,10);
580     writeln(' BILANS ZUZYCIA PALIW ');
581     writeln;
582     TextColor(14);
583     writeln;
584     writeln('                BS                ON');
585     writeln;
586     TextColor(10);
587     writeln('W rejonie wyjściowym pozostało :                ',pMBS:6:2,'                ',pMON:6:2);
588     writeln;
589     MBS:=jnBS#1.3-pMBS;
590     MON:=jnONPK#1.3+jnONPG-pMON;
591     TextColor(14);
592     writein('DO PEWNYCH ZAPASÓW RUCHOMYCH:');
593     writeln;
594     TextColor(15);
```

```
595     writeln('Przy sprzecie nalezy uzupeenia :      ',MBS:6:2,'      ',MON:6:2);
596     writeln;
597     MBS:=jnBS#1.6-pMBS;
598     MON:=jnONPK#1.6+jnONPG#1.6-pMON;
599     TextColor(10);
600     writeln('W brygady nalezy uzupeenia :      ',MBS:6:2,'      ',MON:6:2);
601     czekanie_na_return;
602 end;
```

```
604 Function pobranie_numeru(xpos,ypos : Integer) : Integer;
605 var
606     nr_kol : Integer;
607 begin
608     repeat
609         nr_kol:=0;
610         Gotoxy(xpos,ypos);
611         ClrEol;
612         Readln(num);
613         val(num,nr_kol,k);
614     until k=0;
615     pobranie_numeru:=nr_kol;
616 end;
```

```
618 Procedure wpr_danych;
619 begin
620     wykonano:=true;
621     ClrScr;
622     gotoxy(10,6);
623     TextColor(12);
624     writeln(' WPROWADZENIE DANYCH O SIYACH I SRODKACH GWUMBY MFS ');
625     writeln;
626     TextColor(10);
627     writeln(' Srodek transportu      ilosa szt:      BS      ON');
628     writeln;
629     TextColor(14);
630     nr_linii:=10;
631     for j:=1 to ilosc_sr do
632     begin
633         gotoxy(1,nr_linii);
634         Write(cysterny[j].naz);
635         TextColor(15);
636         ilosc_ben:=pobranie_numeru(50,nr_linii);
637         ilosc_on :=pobranie_numeru(60,nr_linii);
638         cysterny[j].ben:=ilosc_ben;
639         cysterny[j].on :=ilosc_on;
640         cysterny[j].zbn:=0;
641         cysterny[j].zon:=0;
642         ilosc_benzyny(j);
643         nr_linii:=nr_linii+1;
644         TextColor(13);
645         gotoxy(10,22);
646         Write('ilosa BS ',suma_ben:6:1,' t      ilosa ON ',suma_on:6:1,' t');
647         TextColor(14);
648     end;
```

```
649     sa_cysterny:=true;
650 end;

652 Procedure wysw_danych;
653 begin
654     if sa_cysterny then
655     begin
656         ClrScr;
657         gotoxy(20,1);
658         TextColor(12);
659         WriteLn(' SI#Y I BRDOKI S#UJBY MPS ');
660         TextColor(14);
661         WriteLn;
662         WriteLn(' Srodek transportu      iloŝa :      BS      ON');
663         nr_linii:=4;
664         TextColor(10);
665         for k:=1 to ilosc_sr do
666         begin
667             gotoxy(1,nr_linii);
668             Write(cysterny[k].naz);
669             gotoxy(49,nr_linii);
670             if cysterny[k].ben > 0 then
671                 Write(cysterny[k].ben:3)
672             else
673                 Write('---');
674             gotoxy(59,nr_linii);
675             if cysterny[k].on > 0 then
676                 Write(cysterny[k].on:3)
677             else
678                 WriteLn('---');
679             nr_linii:=nr_linii+2;
680         end;
681         WriteLn;
682         writeln;
683         czekanie_na_return;
684         if L1 > 0 then
685         begin
686             ZuBS:=0;
687             ZuON:=0;
688             for k:=1 to ilosc_kol do
689             begin
690                 obi_jed_nap_kol(k);
691                 ZuBS:=ZuBS+KBS*L12*KBS/100;
692                 KONFK:=KONFK*KONFK+KONFG*KONFG;
693                 ZuON:=ZuON+KONFK*L12/100;
694             end;
695             Write(' ');
696             ClrScr;
697             gotoxy(15,5);
698             TextColor(12);
699             WriteLn(' POR#NHANIE POTRZEB Z MO#LIWO#CIAMI ');
700             TextColor(10);
701             writeln;
702             WriteLn('      ZuJycie      Przydzia#');
```

```
703      writeln;
704      writeln('      BS      ON      BS      ON');
705      writeln;
706      TextColor(14);
707      gotoxy(6,11);
708      Write(ZuBS:6:2,' t');
709      gotoxy(20,11);
710      Write(ZuON:6:2,' t');
711      gotoxy(40,11);
712      Write(suma_ben:6:2,' t');
713      gotoxy(52,11);
714      WriteLn(suma_on:6:2,' t');
715      if ((ZuBS > 0) or (ZuON > 0)) then
716      begin
717          TextColor(12);
718          gotoxy(15,16);
719          WriteLn('      POSIADANA ILOSC FALIW STANOWI');
720          writeln;
721          TextColor(10);
722          WriteLn('      BS      ON');
723          TextColor(14);
724          gotoxy(20,20);
725          if (ZuBS > 0) then
726          begin
727              if (ZuBS>0) then
728                  ZuBS:=suma_ben#100/ZuBS
729              else
730                  ZuBS:=0;
731              Write(ZuBS:6:1,' % zuzycia');
732          end
733          else Write('----');
734          gotoxy(39,20);
735          if (ZuON > 0) then
736          begin
737              if (ZuON>0) then
738                  ZuON:=suma_on#100/ZuON
739              else
740                  ZuON:=0;
741              Write(ZuON:6:1,' % zuzycia');
742          end
743          else Write('----');
744          end;
745      end;
746      end
747      else
748      begin
749          wykonano:=false;
750          ClrScr;
751          gotoxy(15,18);
752          WriteLn('Nie przydzielono srodkow sluzby aps',chr(7));
753      end;
754      czekanie_na_return;
755      end;
```

```
757 Procedure podzial_srodkow;
758 begin
759   wykonano:=true;
760   { cysterny }
761   repeat
762     if wykonano then
763       begin
764         ClrScr;
765         GoToxy(12,2);
766         TextColor(12);
767         WriteLn(' ANALIZA I PODZIAŁ POSIADANYCH SIŁ I ŚRODKÓW SŁUŻBY MPS ');
768         TextColor(14);
769         WriteLn;
770         WriteLn('          1. Wprowadzanie danych o siłach i środkach służby mps');
771         WriteLn;
772         WriteLn('          2. Wyświetlenie danych ');
773         WriteLn;
774         WriteLn('          3. Podział sił i środków służby mps');
775         WriteLn;
776         WriteLn('          4. Powrót do głównego menu');
777         WriteLn;
778         WriteLn;
779         TextColor(15+128);
780         Write('          polecenie - ');
781       end
782     else
783       begin
784         for k:=10 to 24 do
785           begin
786             GoToxy(1,k);
787             ClrEol;
788           end;
789       end;
790     gotoxy(43,13);
791     write(' ');
792     GoToxy(43,13);
793     repeat
794       Read(polecenie);
795     until ((polecenie > '0') and (polecenie < '5'));
796     wykonano:=true;
797     case polecenie of
798       '1' : wpr_danych;
799       '2' : begin
800             wysw_danych;
801             podzial_srodkow;
802           end;
803       '3' : podzial;
804     end;
805     until polecenie='4';
806   end;

808 Procedure wczytaj_liczbe(xpos,ypos,dl_licz : Integer ;qgran,qgran : Real; var liczba : Real);
809 begin
810   repeat
```

```
811     GoToxy(xpos,ypos);
812     ClrEol;
813     Readln(number);
814     val(number,liczba,k);
815     until ((k = 0) and (liczba >= dgran) and (liczba <= qgran));
816     end;

818 Function jest_kolumna : Boolean;
819     begin
820     jest_kolumna:=False;
821     for k:=1 to ilosc_kol do
822     if kolumny[k,1].nr_odd > 0 then
823     begin
824     jest_kolumna:=True;
825     exit;
826     end;
827     end;

829 Procedure obliczenie_zuzycia;
830     var
831     droga,sON,sBS,sONPK,sONPG :Real;
832     byla_kol : Boolean;
833     nr_linii : Integer;

835     begin
836     if jest_kolumna then
837     begin
838     repeat
839     ClrScr;
840     GoToxy(15,1);
841     textColor(12);
842     Writeln(' OBLICZANIE ZUZYCIA MPS PRZEZ POSZCZEGOLNE KOLUMNY');
843     TextColor(14);
844     Writeln;
845     Writeln(' Zuzycie bedzie obliczane wedlug nastepujacego wzoru :');
846     Writeln;
847     TextColor(10);
848     Writeln('                L * jn');
849     Writeln('          Zu = K -----');
850     Writeln('                100');
851     TextColor(14);
852     Writeln('Oznaczenia :');
853     writeln('-----');
854     Writeln(' Zu - zuzycie mps w tonach');
855     writeln;
856     Writeln(' L - wielkosc przebytej drogi w marszu');
857     writeln;
858     Writeln(' jn - wielkosc jednostki napelnienia kolumny marszowej w tonach');
859     writeln;
860     Writeln(' K - wspolczynnik zuzycia mps');
861     czekanie_na_return;
862     ClrScr;
863     textColor(12);
864     gotoxy(21,3);
```

```
865 write(' P A R A M E T R Y   W E J S C I O W E ');
866 gotoxy(20,4);
867 write('-----');
868 TextColor(14);
869 gotoxy(9,7);
870 writeln('Podaj wartości współczynników K dla poszczególnych rodzajów paliw');
871 writeln;
872 gotoxy(9,9);
873 write('   KBS = ');
874 wczytaj_liczbe(20,9,4,0,1,KBS);
875 gotoxy(9,10);
876 write('   KONPK = ');
877 wczytaj_liczbe(20,10,4,0,1,KONPK);
878 gotoxy(9,11);
879 write('   KONPG = ');
880 wczytaj_liczbe(20,11,4,0,1,KONPG);
881 writeln;
882 TextColor(10);
883 gotoxy(9,15);
884 write('   M A R S Z ');
885 TextColor(14);
886 gotoxy(9,17);
887 write('Podaj długość marszu do rejonu dwugodzinnego postoju L1 = ');
888 wczytaj_liczbe(67,17,6,0,400,L1);
889 gotoxy(9,18);
890 write('Podaj długość całego marszu           L12 = ');
891 wczytaj_liczbe(67,18,6,L1,400,L12);
892 czekanie_na_return;
893 repeat
894   clrscr;
895   gotoxy(20,2);
896   TextColor(12);
897   writeln(' OBLICZANIE ZUZYCIA NPS W CZASIE MARSZU');
898   TextColor(14);
899   writeln;
900   writeln(' 1. Do rejonu dwugodzinnego postoju');
901   writeln;
902   writeln(' 2. Od rejonu dwugodzinnego postoju do rejonu dziennego/nocnego/ odpoczynku');
903   writeln;
904   writeln(' 3. W czasie całego marszu');
905   writeln;
906   writeln(' 4. Zadanie innych parametrów marszu');
907   writeln;
908   writeln(' 5. Powrót do głównego menu');
909   writeln;
910   writeln;
911   TextColor(15+128);
912   writeln('           Polecenie - ');
913   gotoxy(31,15);
914   write(' ');
915   gotoxy(31,15);
916   repeat
917     read(polecenie);
918   until ((polecenie > '0') and (polecenie < '6'));
```

```

919     if polecenie < '4' then
920         begin
921             sON:=0;
922             sBS:=0;
923             sONPK:=0;
924             sONPG:=0;
925             suma:=0;
926             nr_linii:=4;
927             byla_kol:=false;
928             case polecenie of
929                 '1' : droga:=L1;
930                 '2' : droga:=L12 - L1;
931                 '3' : droga:=L12;
932             end;
933             for k:=1 to ilosc_kol do
934                 begin
935                     if kolumny[k,1].nr_odd > 0 then
936                         begin
937                             obl_jed_nap_kol(k);
938                             ZuBS :=KBS *droga #MBS/100;
939                             ZuONPK:=KONPK*droga#MONPK/100;
940                             ZuONPG:=KONPG*droga#MONPG/100;
941                             sBS:=sBS+ZuBS;
942                             sONPK:=sONPK+ZuONPK;
943                             sONPG:=sONPG+ZuONPG;
944                             byla_kol:=true;
945                             nr_linii:=nr_linii+1;
946                             if nr_linii = 5 then
947                                 begin
948                                     clrscr;
949                                     TextColor(12);
950                                     gotoxy(1,1);
951                                     case polecenie of
952                                         '1' : begin
953                                             writeln('          ZUZYCIE MPS NA DRODZE DO REJONU DWUGODZINNEGO POSTOJ
954                                             U');
955                                             end;
956                                         '2' : begin
957                                             writeln('          ZUZYCIE MPS NA DRODZE OD REJONU POSTOJU DO REJONU DZIENNEG
958                                             O ODPOCZYNKU');
959                                             end;
960                                         '3' : begin
961                                             writeln('          ZUZYCIE MPS W CZASIE CAPEGO MARSZU');
962                                             end;
963                                     end;
964                                     writeln;
965                                     TextColor(10);
966                                     write(' Kolumna ');
967                                     write(' ');
968                                     write(' ZuBS   ZuONPK   ZuONPG   sumaON');
969                                     gotoxy(62,5);
970                                     TextColor(10);
971                                     write(' KBS = ',KBS:4:2);
972                                     gotoxy(62,6);

```

```
971         write('KONPK = ',KONPK:4:2);
972         gotoxy(62,7);
973         write('KONPG = ',KONPG:4:2);
974         gotoxy(62,10);
975         write('L1 = ',L1:5:1,' ka');
976         gotoxy(62,11);
977         write('L2 = ',L12-L1:5:1,' ka');
978         gotoxy(62,12);
979         write('L12 = ',L12:5:1,' ka');
980     end;
981     TextColor(15);
982     gotoxy(5,nr_linii);
983     write(k:2);
984     gotoxy(12,nr_linii);
985     if ZuBS > 0 then
986         write(ZuBS:6:2) else write('-----');
987     gotoxy(22,nr_linii);
988     if ZuONPK > 0 then
989         write(ZuONPK:6:2) else write('-----');
990     gotoxy(32,nr_linii);
991     if ZuONPG > 0 then
992         write(ZuONPG:6:2) else write('-----');
993     suma:=ZuONPK+ZuONPG;
994     gotoxy(42,nr_linii);
995     if suma > 0 then
996         write(suma:6:2) else write('-----');
997     end;
998     end;
999     TextColor(10);
1000     writeln;
1001     writeln('-----');
1002     TextColor(12);
1003     write(' suma');
1004     gotoxy(11,nr_linii+2);
1005     if (sBS > 0) then write(sBS:7:2) else write('-----');
1006     gotoxy(21,nr_linii+2);
1007     if (sONPK > 0) then write(sONPK:7:2) else write('-----');
1008     gotoxy(31,nr_linii+2);
1009     if (sONPG > 0) then write(sONPG:7:2) else write('-----');
1010     gotoxy(41,nr_linii+2);
1011     sON:=sONPK+sONPG;
1012     if (sON > 0) then write(sON:7:2) else write('-----');
1013     czekanie_na_return;
1014     end;
1015     until ((polecenie = '4') or (polecenie = '5'));
1016     until polecenie = '5';
1017     end
1018     else
1019     begin
1020         ClrScr;
1021         gotoxy(19,18);
1022         writeln('Nie stworzono kolumn',chr(7));
1023         czekanie_na_return;
1024         nie_wym:=false;
```

```
1025     end;
1026 end;

1028 Procedure obliczenie_masy_in;
1029 var
1030     byla_kolumna : Boolean;
1031     nr_od,nr_linii,akt_oddzial: integer;
1032     pr : real;
1033 begin
1034     nr_linii:=3;
1035     byla_kolumna:=false;
1036     for k:=1 to ilosc_kol do
1037         begin
1038             MBS:=0;
1039             MONPK:=0;
1040             MONPG:=0;
1041             if kolumny[k,1].nr_odd > 0 then
1042                 begin
1043                     if nr_linii = 3 then ClrScr;
1044                     nr_linii:=nr_linii+2;
1045                     byla_kolumna:=true;
1046                     akt_oddzial:=1;
1047                     while kolumny[k,akt_oddzial].nr_odd > 0 do
1048                         begin
1049                             nr_od:=kolumny[k,akt_oddzial].nr_odd;
1050                             ul:=kolumny[k,akt_oddzial].ulamiek;
1051                             pr:=kolumny[k,akt_oddzial].proc/100.0;
1052                             with tab_jed_nap[nr_od] do
1053                                 begin
1054                                     MBS:= MBS+ BS*ul*pr;
1055                                     MONPK:=MONPK+ONPK*ul*pr;
1056                                     MONPG:=MONPG+ONPG*ul*pr;
1057                                 end;
1058                             akt_oddzial:=akt_oddzial+1;
1059                         end;
1060                     TextColor(15);
1061                     gotoxy(3,nr_linii);
1062                     write(' ',k);
1063                     gotoxy(18,nr_linii);
1064                     if MBS > 0 then write(MBS:6:2)
1065                         else write('-----');
1066                     gotoxy(26,nr_linii);
1067                     if MONPK > 0 then write(MONPK:6:2)
1068                         else write('-----');
1069                     gotoxy(36,nr_linii);
1070                     if MONPG > 0 then write(MONPG:6:2)
1071                         else write('-----');
1072                     gotoxy(47,nr_linii);
1073                     suma:=MONPK+MONPG;
1074                     if suma > 0 then write(suma:6:2)
1075                         else write('-----');
1076                 end;
1077             end;
1078             if not byla_kolumna then
```

```
1079     begin
1080         nie_wym:=true;
1081         gotoxy(20,20);
1082         write(' Nie sformowano kolumn ',chr(7));
1083     end
1084 else
1085     begin
1086         TextColor(12);
1087         gotoxy(9,1);
1088         writeln(' ZESTAWIENIE MASY JEDNOSTK NAPEWNIENIA KOLUMN ');
1089         Textcolor(14);
1090         writeln;
1091         write(' Kolumny ');
1092         gotoxy(16,3);
1093         writeln('   BS   ONPK   ONPG   sumaON ');
1094         TextColor(15);
1095         czekanie_na_return;
1096     end;
1097 end;
```

```
1099 Procedure zapis_jedn_nap;
```

```
1100     var
```

```
1101         JNAP : file of jedn_nap;
```

```
1103     begin
```

```
1104         Assign(JNAP,'jedn_nap');
```

```
1105         {$I-}
```

```
1106         rewrite(JNAP);
```

```
1107         {$I+}
```

```
1108         if (IORResult=0) then
```

```
1109             begin
```

```
1110                 for k:=1 to ilosc_podod do
```

```
1111                     Write(JNAP,tab_jed_nap[k]);
```

```
1112                 Close(JNAP);
```

```
1113             end
```

```
1114         else
```

```
1115             begin
```

```
1116                 gotoxy(22,15);
```

```
1117                 Writeln('Brak zbioru JEDN_NAP',chr(7));
```

```
1118                 czekanie_na_return;
```

```
1119             end;
```

```
1120     end;
```

```
1121 Procedure zapis_cystern;
```

```
1122     var
```

```
1123         CYST : file of opis_cyst;
```

```
1124     begin
```

```
1125         Assign(CYST,'cysterny');
```

```
1126         {$I-}
```

```
1127         Rewrite(CYST);
```

```
1128         {$I+}
```

```
1129         if (IGResult=0) then
```

```
1130             begin
```

```
1131                 for k:=1 to ilosc_sr do
```

```
1132                     Write(CYST,cysterny[k]);
```

```
1133     Close(CYST);
1134     end;
1135     end;
1136 Procedure odczyt_cystern;
1137     var
1138     CYST : file of opis_cyst;
1139     begin
1140     Assign(CYST,'cysterny');
1141     (%I-)
1142     Reset(CYST);
1143     (%I+)
1144     if (IOResult=0) then
1145     begin
1146     ClrScr;
1147     gotoxy(20,1);
1148     TextColor(12);
1149     WriteLn(' SIYY I SR00KI SWU4BY MPS ');
1150     TextColor(14);
1151     WriteLn;
1152     WriteLn('         Srodek transportu         ilosc :  BS         GW');
1153     TextColor(10);
1154     nr_linii:=4;
1155     for k:=1 to ilosc_sr do
1156     begin
1157     read(CYST,cysterny[k]);
1158     gotoxy(1,nr_linii);
1159     Write(cysterny[k].naz);
1160     gotoxy(49,nr_linii);
1161     if cysterny[k].ben > 0 then
1162     Write(cysterny[k].ben:3)
1163     else
1164     Write('---');
1165     gotoxy(59,nr_linii);
1166     if cysterny[k].on > 0 then
1167     Write(cysterny[k].on:3)
1168     else
1169     WriteLn('---');
1170     nr_linii:=nr_linii+2;
1171     end;
1172     WriteLn;
1173     writeln;
1174     czekanie_na_return;
1175     Close(CYST);
1176     end
1177     else
1178     begin
1179     ClrScr;
1180     gotoxy(22,15);
1181     writeln('Brak zbioru CYSTERNY',chr(7));
1182     czekanie_na_return;
1183     end;
1184     end;
1185 Procedure odczyt_jedn_nap;
1186     var
```

```
1187     JNAP : file of jedn_nap;
1188     n     : integer;
1189 begin
1190     Assign(JNAP,'jedn_nap');
1191     (¶I-)
1192     Reset(JNAP);
1193     (¶I+)
1194     if (IORResult=0) then
1195         begin
1196             ClrScr;
1197             TextColor(12);
1198             gotoxy(5,2);
1199             writeln('TABELA JEDNOSTEK NAPEWNIENIA');
1200             writeln;
1201             TextColor(14);
1202             writeln(' Nr pododdz  BS  ONPK  ONPG  ');
1203             n:=4;
1204             TextColor(15);
1205             for k:=1 to ilosc_podod do
1206                 begin
1207                     Read(JNAP,tab_jed_nap[k]);
1208                     gotoxy(6,n+2);
1209                     writeln( k);
1210                     gotoxy(14,n+2);
1211                     if tab_jed_nap[k].BS > 0 then
1212                         write(tab_jed_nap[k].BS:4:2)
1213                     else
1214                         Write('----');
1215                     GoToxy(20,n+2);
1216                     if tab_jed_nap[k].ONPK > 0 then
1217                         Write(tab_jed_nap[k].ONPK:4:2)
1218                     else
1219                         Write('----');
1220                     GoToxy(27,n+2);
1221                     if tab_jed_nap[k].ONPG > 0 then
1222                         Write(tab_jed_nap[k].ONPG:4:2)
1223                     else
1224                         Write('----');
1225                     n:=n+1;
1226                 end;
1227             Close(JNAP);
1228             czekanie_na_return;
1229         end
1230     else
1231         begin
1232             ClrScr;
1233             gotoxy(22,15);
1234             writeln('Brak zbioru JED_NAP',chr(7));
1235             czekanie_na_return;
1236         end;
1237     end;

1239 Procedure wysw_oddzial(nr_kol,akt_oddzial : Integer);
1240 - var
```

```
1241   nr_linii, pos_w_linii : Integer;
1242   begin
1243   if kolumny[nr_kol,akt_oddzial].nr_odd <> 0 then
1244   begin
1245   nr_linii:=(akt_oddzial-1) div 3;
1246   pos_w_linii:=(akt_oddzial-1) mod 3;
1247   Gotoxy(pos_w_linii*27+1,nr_linii+15);
1248   if kolumny[nr_kol,akt_oddzial].ulamek < 1 then
1249   Write(kolumny[nr_kol,akt_oddzial].wk_odd);
1250   Write(nazwa_podod[kolumny[nr_kol,akt_oddzial].nr_odd]);
1251   end;
1252   end;

1254 Procedure odczyt_skl_kol;
1255   var
1256   SKOL : file of skl_kol;
1257   nr_kol : integer;
1258   akt_oddzial : integer;
1259   begin
1260   Assign(SKOL,'kolumny');
1261   (#I-)
1262   Reset(SKOL);
1263   (#I+)
1264   if (IOResult=0) then
1265   begin
1266   ClrScr;
1267   k:=1;
1268   repeat
1269   nr_kol:=k;
1270   for j:=1 to ilosc_podod do
1271   Read(SKOL,kolumny[k,j]);
1272   if kolumny[nr_kol,1].nr_odd <> 0 then
1273   begin
1274   TextColor(14);
1275   gotoxy(15,13);
1276   writeln(' SK#AD KOLUMNY NR ',k);
1277   TextColor(15);
1278   for akt_oddzial:=1 to ilosc_podod do
1279   if kolumny[nr_kol,akt_oddzial].nr_odd <> 0 then
1280   begin
1281   wysw_oddzial(nr_kol,akt_oddzial);
1282   end;
1283   czekanie_na_return;
1284   end;
1285   ClrScr;
1286   k:=k+1;
1287   until k > ilosc_kol;
1288   Close(SKOL);
1289   end
1290   else
1291   begin
1292   ClrScr;
1293   gotoxy(22,15);
1294   WriteLn('Brak zbioru KOLUMNY',chr(7));
```

```
1295     czekanie_na_return;
1296     end;
1297 end;

1300 Procedure zapis_ski_kol;
1301 var
1302     SKOL : file of ski_kol;
1303 begin
1304     Assign(SKOL,'kolumny');
1305     {$I-}
1306     Rewrite(SKOL);
1307     {$I+}
1308     if (IOResult=0) then
1309     begin
1310         for k:=1 to ilosc_kol do
1311             for j:=1 to ilosc_podod do
1312                 Write(SKOL,kolumny[k,j]);
1313             Close(SKOL);
1314         end;
1315     end;
1316 Procedure operacje_dyskowe;
1317 begin
1318     ClrScr;
1319     gotoxy(24,2);
1320     TextColor(12);
1321     Writeln(' OPERACJE DYSKOWE ');
1322     TextColor(14);
1323     writeln;
1324     Writeln('         1. Zapis tabeli jednostek nape@nienia');
1325     writeln;
1326     Writeln('         2. Odczyt tabeli jednostek nape@nienia');
1327     writeln;
1328     Writeln('         3. Zapis sk@adu kolumn');
1329     writeln;
1330     Writeln('         4. Odczyt sk@adu kolumn');
1331     writeln;
1332     Writeln('         5. Zapis srodk@w s@usby mps');
1333     writeln;
1334     Writeln('         6. Odczyt srodk@w s@usby mps');
1335     writeln;
1336     Writeln('         7. Powr@t do g@@wnego menu');
1337     writeln;
1338     TextColor(15+128);
1339     write ('         polecenie - ');
1340     repeat
1341         for k:=19 to 24 do
1342             begin
1343                 gotoxy(1,k);
1344                 ClrEol;
1345             end;
1346     gotoxy(38,18);
1347     write(' ');
1348     gotoxy(38,18);
```

```
1349 repeat
1350     read(polecenie);
1351 until ((polecenie > '0') and (polecenie < '8'));
1352 case polecenie of
1353     '1' : zapis_jedn_nap;
1354     '2' : begin
1355         odczyt_jedn_nap;
1356         operacje_dyskowe;
1357     end;
1358     '3' : zapis_skl_kol;
1359     '4' : begin
1360         odczyt_skl_kol;
1361         operacje_dyskowe;
1362     end;
1363     '5' : zapis_cystrern;
1364     '6' : begin
1365         odczyt_cystrern;
1366         operacje_dyskowe;
1367     end;
1368 end;
1369 until polecenie = '7';
1370 end;

1374 Function cyfra( znak : char) : boolean;
1375 begin
1376     cyfra:=false;
1377     if ((znak >= '0') and (znak <= '9')) then cyfra:=true;
1378 end;

1380 Procedure wysw_nazwy;
1381 var
1382     i,j : Integer;
1383 begin
1384     for k:=1 to 10 do
1385         begin
1386             GoToxy(1,k);
1387             ClrEol;
1388         end;
1389     Clrscr;
1390     GoToxy(2,1);
1391     TextColor(12);
1392     WriteLn('          DOPUSZCZALNE    NAZWY    PODODDZIA#a# ');
1393     ClrEol;
1394     TextColor(10);
1395     WriteLn;
1396     ClrEol;
1397     for k:=1 to ilosc_podod do
1398         begin
1399             i:=(k-1) div 7;
1400             j:=(k-1) mod 7;
1401             GoToxy(j#12+1,i+3);
1402             Write(nazwa_podod[k]);
```

```
1403     end;
1404 end;

1406 function oblicz_numer(naz : nazwa) : integer;
1407 begin
1408   if naz[0]=chr(0) then
1409     begin
1410       oblicz_numer:=100;
1411       exit;
1412     end;
1413   for k:=1 to ilosc_podod do
1414     begin
1415       if ((pos(naz,nazwa_podod[k]) <> 0) and (pos(nazwa_podod[k],naz) <> 0))then
1416         begin
1417           oblicz_numer:=k;
1418           exit;
1419         end;
1420     end;
1421   oblicz_numer:=0;
1422 end;

1424 Procedure utw_kolumny;
1425 var
1426   numer_odd,nr_kol,akt_oddzial : integer;
1427   procent : real;

1429 begin
1430   ClrScr;
1431   TextColor(15);
1432   repeat
1433     gotoxy(1,12);
1434     write(' Podaj numer kolumny : ');
1435     nr_kol:=pobranie_numeru(25,12);
1436   until ((nr_kol > 0) and (nr_kol <= ilosc_kol));
1437   writeln;
1438   wysw_nazwy;
1439   gotoxy(1,8);
1440   TextColor(14);
1441   writeln(' Kolumna nr : ',nr_kol);
1442   textcolor(15);
1443   writeln;
1444   writeln(' #prowadź kolejną nazwę pododdziału: ');
1445   akt_oddzial := 1;
1446   repeat
1447     gotoxy(40,12);
1448     ClrEol;
1449     gotoxy(40,10);
1450     ClrEol;
1451     readln(nazwa_odd);
1452     if cyfra(nazwa_odd[1]) and cyfra(nazwa_odd[3]) and (nazwa_odd[2] = '/') then
1453       begin
1454         kolumny[nr_kol,akt_oddzial].wkl_odd:=copy(nazwa_odd,1,3);
1455         cyfra_1:=ord(nazwa_odd[1])-48;
1456         cyfra_2:=ord(nazwa_odd[3])-48;
```

```
1457     kolumny[nr_kol,akt_oddzial].ulamek:=cyfra_1/cyfra_2;
1458     nazwa_odd:=copy(nazwa_odd,4,10);
1459     end
1460   else
1461     kolumny[nr_kol,akt_oddzial].ulamek:=1;
1462     numer_odd:=oblicz_numer(nazwa_odd);
1463     if ((numer_odd > 0) and (numer_odd <= ilosc_podod)) then
1464       begin
1465         kolumny[nr_kol,akt_oddzial].nr_odd:=numer_odd;
1466         repeat
1467           writeln;
1468           writeln (' Podaj procent ukończenia : ');
1469           procent:=100;
1470           gotoxy(40,12);
1471           ClrEol;
1472           readln(procent);
1473           until ((procent >= 0) and (procent <= 100));
1474           kolumny[nr_kol,akt_oddzial].proc:=procent;
1475           wysw_oddzial(nr_kol,akt_oddzial);
1476           akt_oddzial:=akt_oddzial+1;
1477         end;
1478       until ((numer_odd = 100) or (akt_oddzial > ilosc_podod));
1479       for k:=akt_oddzial to ilosc_podod do
1480         kolumny[nr_kol,k].nr_odd:=0;
1481     end;

1483 Procedure czytaj_liczbe(xpos,ypos : Integer; var Zmienna : Real);
1484   var
1485     code : Integer;
1486     liczba : string[6];
1487   begin
1488     repeat
1489       GoToxy(xpos,ypos);
1490       ClrEol;
1491       Readln(liczba);
1492       Val(liczba,Zmienna,code);
1493     until code=0;
1494   end;

1496 Procedure zestawienie_masy;
1497   var
1498     n : integer;
1499   begin
1500     repeat
1501       ClrScr;
1502       TextColor(12);
1503       GoToxy(1,1);
1504       Writeln(' ZESTAWIENIE MASY JEDNOSTEK NAPEWNIENIA PODODZIAŁÓW BRYGADY');
1505       TextColor(14);
1506       writeln;
1507       Write(' Oddzial ');
1508       NoreVideo;
1509       Write(' ');
1510       Write(' BS   ONPK   ONFG   suma ON ');
```

```
1511 TextColor(10);
1512 writeln;
1513 n:=3;
1514 for k:=1 to ilosc_podod do
1515   begin
1516     gotoxy(2,n+2);
1517     Write(nazwa_podod[k]);
1518     GoToxy(16,n+2);
1519     if tab_jed_nap[k].BS > 0 then
1520       Write(tab_jed_nap[k].BS:4:2)
1521     else
1522       Write('----');
1523     GoToxy(23,n+2);
1524     if tab_jed_nap[k].ONPK > 0 then
1525       Write(tab_jed_nap[k].ONPK:4:2)
1526     else
1527       Write('----');
1528     GoToxy(32,n+2);
1529     if tab_jed_nap[k].ONPG > 0 then
1530       Write(tab_jed_nap[k].ONPG:4:2)
1531     else
1532       Write('----');
1533     GoToxy(41,n+2);
1534     suma:=tab_jed_nap[k].ONPK+tab_jed_nap[k].ONPG;
1535     if suma > 0 then
1536       Write(suma:5:2)
1537     else
1538       Write('----');
1539     writeln;
1540     n:=n+1;
1541   end;
1542 writeln;
1543 TextColor(15);
1544 Write(' Czy chcesz zmienic jn pododdzialu (t/n) ? ');
1545 repeat
1546   Read(polecenie);
1547   polecenie:=UpCase(polecenie);
1548 until ((polecenie='T') or (polecenie='N'));
1549 if polecenie='T' then
1550   begin
1551     ClrScr;
1552     GoToxy(10,10);
1553     Write('Podaj nazwe oddzialu');
1554     repeat
1555       GoToxy(40,10);
1556       ClrEol;
1557       ReadLn(Nazwa_odd);
1558       numer_odd:=oblicz_numer(Nazwa_odd);
1559     until ((numer_odd > 0) and (numer_odd < ilosc_podod));
1560     GoToxy(10,12);
1561     Write('Podaj nowe jn :');
1562     GoToxy(15,13);
1563     Write('BS ');
1564     czytaj_liczbe(21,13,tab_jed_nap[numer_odd].BS);
```

```
1565      GoToxy(15,14);
1566      Write('GNPK ');
1567      czytaj_liczbe(21,14,tab_jed_nap[numer_odd].GNPK);
1568      GoToxy(15,15);
1569      Write('GNPG ');
1570      czytaj_liczbe(21,15,tab_jed_nap[numer_odd].GNPG);
1571      end;
1572  until polecenie='N';
1573  end;
```

```
1575 Procedure przegladanie_kolumn;
1576 var
1577   nr_kol,akt_oddzial : Integer;
1578   byla_kol           : Boolean;

1580 begin
1581   byla_kol:=False;
1582   ClrScr;
1583   GoToxy(25,10);
1584   Write('Podaj numer kolumny ');
1585   repeat
1586     nr_kol:=pobranie_numeru(47,10);
1587     until ((nr_kol > 0) and (nr_kol <=ilosc_kol));
1588   ClrScr;
1589   if kolumny[nr_kol,1].nr_odd<>0 then
1590     for akt_oddzial:=1 to ilosc_podod do
1591       if kolumny[nr_kol,akt_oddzial].nr_odd<>0 then
1592         begin
1593           wysw_oddzial(nr_kol,akt_oddzial);
1594           byla_kol:=True;
1595         end;
1596   if byla_kol then
1597     begin
1598       GoToxy(25,12);
1599       TextColor(12);
1600       Writeln(' SKLAD KOLUMNY NR ',nr_kol,' ');
1601       TextColor(14);
1602     end
1603   else
1604     begin
1605       GoToxy(18,18);
1606       Write(' Nie sformowano kolumny ',nr_kol,chr(7));
1607     end;
1608   czekanie_na_return;
1609 end;
```

```
1611 Procedure menu_form;
1612 begin
1613   for k:=1 to 10 do
1614     begin
1615       GoToxy(1,k);
1616       ClrEol;
1617     end;
1618   TextColor(12);
```

```
1619   GoToxy(20,3);
1620   WriteLn(' FORMOWANIE KOLUMN ');
1621   TextColor(14);
1622   GoToxy(20,5);
1623   Write(' 1. Przeglądanie istniejących kolumn');
1624   GoToxy(20,7);
1625   Write(' 2. Sformowanie nowej kolumny');
1626   GoToxy(20,9);
1627   Write(' 3. Powrót do głównego menu');
1628   TextColor(15+128);
1629   GoToxy(32,11);
1630   Write('Polecenie - ');
1631   end;

1633 Procedure sformowanie_kolumn;
1634   begin
1635     ClrScr;
1636     menu_form;
1637     repeat
1638       for k:=12 to 24 do
1639         begin
1640           GoToxy(1,k);
1641           ClrEol;
1642         end;
1643       gotoxy(44,11);
1644       write(' ');
1645       GoToxy(44,11);
1646       repeat
1647         Read(polecenie);
1648         polecenie:=UpCase(polecenie);
1649         until ((polecenie > '0') and (polecenie < '4'));
1650       case polecenie of
1651         '1' : begin
1652             przegladanie_kolumn;
1653             menu_form;
1654           end;
1655         '2' : begin
1656             utw_kolumna;
1657             menu_form;
1658           end;
1659       end;
1660       until (polecenie = '3');
1661     end;

1663 Procedure koniec_pracy;
1664   begin
1665     ClrScr;
1666     Halt;
1667   end;

1669 Procedure wyswietlenie_menu;
1670   begin
1671     ClrScr;
1672     TextColor(12);
```

```
1673 GoToxy(8,3);
1674 Writeln(' PROGRAM ANALIZY ZUZYCIA I SPOSOBU PRZYDZIAWU MPS ');
1675 GoToxy(2,5);
1676 TextColor(10);
1677 Writeln(' Główny menu : ');
1678 gotoxy(2,6);
1679 writeln(' ----- ');
1680 TextColor(14);
1681 Writeln;
1682 Writeln('      1. Zestawienie masy jednostek napełnienia pododdziałów');
1683 writeln;
1684 Writeln('      2. Sformowanie kolumny marszowej');
1685 writeln;
1686 Writeln('      3. Obliczenie jednostek napełnienia poszczególnych kolumn');
1687 writeln;
1688 Writeln('      4. Obliczenie zużycia mps przez poszczególne kolumny');
1689 writeln;
1690 Writeln('      5. Podział posiadanych sił i środków służby mps');
1691 writeln;
1692 Writeln('      6. Operacje dyskowe');
1693 writeln;
1694 Writeln('      7. Zakończenie pracy z programem');
1695 Writeln;
1696 writeln;
1697 writeln;
1698 TextColor(15+128);
1699 Write('                Polecenie - ');
1700 end;

1702 Procedure inicjalizacja(odd_nr : Integer; oddzial : Nazwa; jnBS,jnONPK,jnONFG : Real);
1703 begin
1704     nazwa_podod[odd_nr]:=oddzial;
1705     with tab_jed_nap[odd_nr] do
1706     begin
1707         BS:=jnBS;
1708         ONPK:=jnONPK;
1709         ONFG:=jnONFG;
1710     end;
1711 end;
1712 begin
1713     TextColor(15);
1714     sa_cysterny:=False;
1715     cysterny[1].naz:= 'samochod transportowy a beczkami STRB';
1716     cysterny[1].pojem:=3.6;
1717     cysterny[2].naz:= 'cysterna dystrybutor 4.5      CD4.5';
1718     cysterny[2].pojem:=4.5;
1719     cysterny[3].naz:= 'cysterna dystrybutor 5      CD5W';
1720     cysterny[3].pojem:=5;
1721     cysterny[4].naz:= 'cyst. dyst. paliwowo-olejowy CDPO4W';
1722     cysterny[4].pojem:=4;
1723     cysterny[5].naz:= 'cysterna dystrybutor 7.5      CD7.5';
1724     cysterny[5].pojem:=7.5;
1725     cysterny[6].naz:= 'samochod transportowy z kanistr. STRK';
1726     cysterny[6].pojem:=3.6;
```

```

1727   cystermy[7].naz:='cysterna dystrybutor      CD5WK';
1728   cystermy[7].pojem:=4.5;
1729   cystermy[8].naz:='cysterna przyczepa        CP';
1730   cystermy[8].pojem:=3;
1731   cystermy[9].naz:='przyczepa transportowa z beczkami PT';
1732   cystermy[9].pojem:=3;
1733   cystermy[10].naz:='cysterna z przyczepa     CD7.5+CP11';
1734   cystermy[10].pojem:=18.5;
1735   for k:=1 to ilosc_kol do
1736     kolumny[k,1].nr_odd:=0;
1737   ClrScr;
1738   TextColor(14);
1739   HighVideo;
1740   GoToxy(5,3);
1741   Write(chr($c9));
1742   for k:=1 to 70 do Write(chr($cd));
1743   WriteLn(chr($bb));
1744   WriteLn(' ',chr($ba),' ',chr($ba)
1745   ));
1745   WriteLn(' ',chr($ba),' ',chr($ba)
1746   ));
1746   WriteLn(' ',chr($ba),'          CENTRUM INFORMATYKI AON ',chr($ba)
1747   ));
1747   WriteLn(' ',chr($ba),'          KATEDRY LOGISTYKI WYDZIAŁU WOJSK LĄDOWYCH AON ',chr($ba)
1748   ));
1748   WriteLn(' ',chr($ba),' ',chr($ba)
1749   ));
1749   WriteLn(' ',chr($ba),' ',chr($ba)
1750   ));
1750   WriteLn(' ',chr($ba),' ',chr($ba)
1751   ));
1751   WriteLn(' ',chr($ba),' ',chr($ba)
1752   ));
1752   TextColor(10);
1753   WriteLn(' ',chr($ba),' prezentuje program : ',chr($ba)
1754   ));
1754   WriteLn(' ',chr($ba),' ',chr($ba)
1755   ));
1755   TextColor(12);
1756   WriteLn(' ',chr($ba),' ',chr($ba)
1757   ));
1757   WriteLn(' ',chr($ba),'          ZAOPATRYWANIE W MFS BRYGADY ZMECHANIZOWANEJ ',chr($ba)
1758   ));
1758   WriteLn(' ',chr($ba),'          WYKONUJĄCEJ MARSZ ',chr($ba)
1759   ));
1759   WriteLn(' ',chr($ba),' ',chr($ba)
1760   ));
1760   TextColor(14);
1761   WriteLn(' ',chr($ba),' ',chr($ba)
1762   ));
1762   WriteLn(' ',chr($ba),' ',chr($ba)
1763   ));
1763   TextColor(10);
1764   WriteLn(' ',chr($ba),' opracowany przez : ',chr($ba)

```

```

));
1765 WriteLn(' ',chr($ba),' ',chr($ba)
));
1766 TextColor(14);
1767 WriteLn(' ',chr($ba),' p&ek dr inz. Zbigniew KLIMKIEWICZ ',chr($ba)
));
1768 WriteLn(' ',chr($ba),' p&ek dr Aleksander #OJKO ',chr($ba)
));
1769 WriteLn(' ',chr($ba),' ',chr($ba)
));
1770 Write(' ',chr($c8));
1771 for k:=1 to 70 do Write(chr($cd));
1772 WriteLn(chr($bc));
1773 WriteLn;
1774 TextColor(15+128);
1775 Write(' ',chr($c8),' Nacisnij klawisz ENTER');
1776 Read(polecenie);
1777 inicjalizacja( 1,'bcz/T-72/', 1.27, 0.80,34.02);
1778 inicjalizacja( 2,'bcz/T-55/', 1.27, 0.80,27.30);
1779 inicjalizacja( 3,'bz', 2.46, 1.52, 0.80);
1780 inicjalizacja( 4,'bpz&ot', 2.46, 2.10, 0.46);
1781 inicjalizacja( 5,'d&ah', 2.60, 1.36, 6.67);
1782 inicjalizacja( 6,'d&aplot', 2.05, 2.86, 2.05);
1783 inicjalizacja( 7,'b&appanc', 1.81, 0, 0);
1784 inicjalizacja( 8,'b&down', 2.46, 3.10, 1.10);
1785 inicjalizacja( 9,'kr', 1.30, 0, 0);
1786 inicjalizacja(10,'ks&ap', 1.04, 1.39, 4.22);
1787 inicjalizacja(11,'k&med', 1.04, 0, 0);
1788 inicjalizacja(12,'k&zaop', 8.04, 7.56, 0);
1789 inicjalizacja(13,'k&rew', 8.35, 1.66, 2.66);
1790 nie_wym:=False;
1791 repeat
1792 if not nie_wym then wyswietlenie_menu
1793 else
1794 begin
1795 for k:=16 to 24 do
1796 begin
1797 GoToxy(1,k);
1798 ClrEol;
1799 end;
1800 end;
1801 GoToxy(38,24);
1802 repeat
1803 Read(polecenie);
1804 until ((polecenie > '0') and (polecenie < '8'));
1805 nie_wym:=False;
1806 case polecenie of
1807 '1' : zestawienie_masy;
1808 '2' : sformowanie_kolumn;
1809 '3' : obliczenie_masy_in;
1810 '4' : obliczenie_zuzycia;
1811 '5' : podzial_srodkow;
1812 '6' : operacje_dyskowe;
1813 '7' : koniec_pracy;

```

1814 end;
1815 until False;
1816 end.

