

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

JAWNE

WYDZIAŁ WOJSK LĄDOWYCH - KATEDRA LOGISTYKI



ZASTRZEŻONE
POUFNE

Egz. Nr ..1



ZAOPATRYWANIE W MPS PUŁKU ZMECHANIZOWANEGO

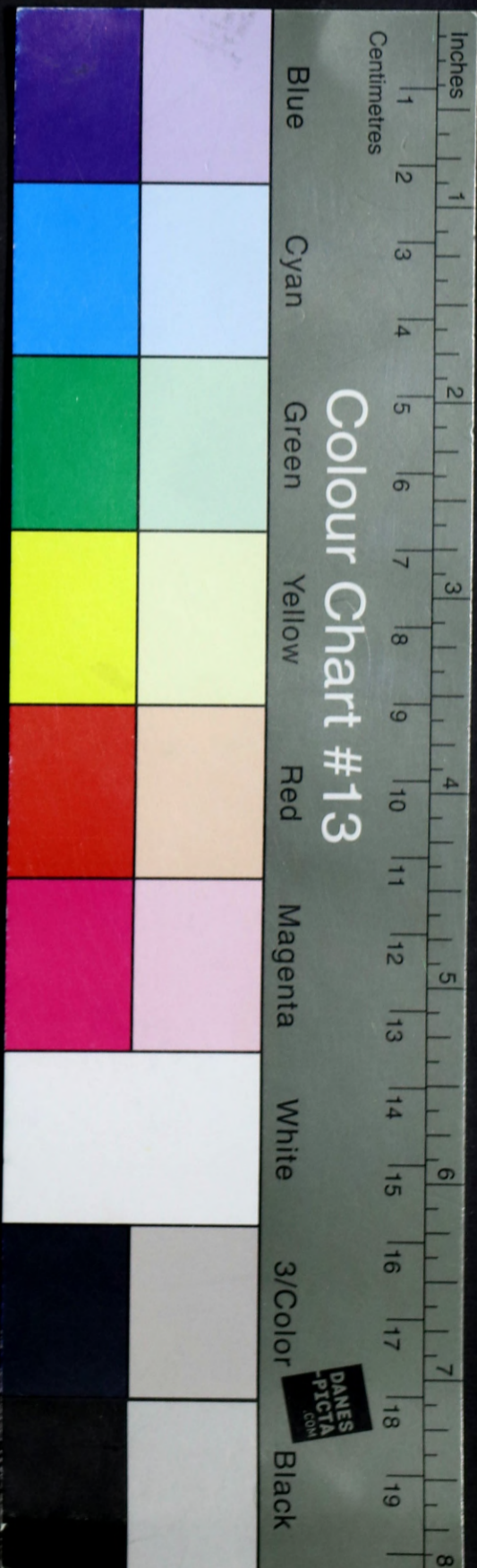
WYKONUJĄCEGO MARSZ



60748

WARSZAWA

1993



AKADEMIA OBRONY NARODOWEJ

Centrum Informatyki
Wydział Wojsk Lądowych - Katedra Logistyki

JAWNE

160206 Anna KOLEK
Podst. prot. przed. Nvuch 648
z olh. 24022006



**ZASTRZEŻONE
POUFNE**

Exz. nr ... 1



ZACPATRYWANIE W MPS PUŁKU ZMECHANIZOWANEGO WYKONUJĄCEGO MARSZ

~~Dr. Kucharski
Opracowanie historii me 20000715201E
29.01.1993 Jan Kucharski, etc.~~

Opracował zespół w składzie:

płk dr hab. inż. Czesław FLANEK
ppłk dr inż. Zbigniew KLIMKIEWICZ
płk dr Aleksander ŁOJKO

WARSZAWA

1993

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...	8
2. ZABEZPIECZENIE MATERIAŁOWE BATALIONU I PUŁKU W MARSZU.....	10
3. UWARUNKOWANIA TAKTYCZNO-LOGISTYCZNE MAJĄCE WPŁYW NA WIELKOŚĆ POTRZEB PALIW PŁYNNYCH W MARSZU.....	12
4. METODY OKREŚLANIA ZUŻYCIA MPS W MARSZU	15
4.1. Metoda statystyczna.....	15
4.2. Metoda matematyczna.....	15
4.2.1. Współczynnik eksploatacyjny k_e	16
4.2.2. Współczynnik manewrowy k_{mt}	16
4.3. Obliczanie zużycia MPS przez poszczególne kolumny.....	18
5. WZORY MATEMATYCZNE DO OBLICZANIA PARAMETRÓW OPERACYJNO - LOGISTYCZNYCH W ZAKRESIE MPS.....	19
6. OPIS PROGRAMU "ZAOPATRYWANIE W MPS pz W MARSZU" (MPSMARSZ).....	21
6.1. Przeznaczenie programu.....	21
6.2. Uruchomienie programu.....	21
6.3. Obsługa programu.....	21
6.3.1. Zestawienie masy jednostek napełnienia pododdziałów.....	23
6.3.2. Sformowanie kolumny marszowej.....	24
6.3.2.1. Sformowanie nowej kolumny.....	25
6.3.2.2. Przeglądanie istniejących kolumn.....	27
6.3.3. Obliczanie masy jednostek napełnienia kolumn marszowych.....	27
6.3.4. Obliczanie zużycia MPS przez poszczególne kolumny.....	28
6.3.5. Podział posiadanych sił i środków służby MPS.....	31
6.3.5.1. Wprowadzenie danych o siłach i środkach służby MPS.....	31
6.3.5.2. Wyświetlanie danych.....	32
6.3.5.3. Podział sił i środków służby MPS.....	33

6.3.6.	Operacje dyskowe.....	35
6.3.6.1.	Odczyt tabeli jednostek napełnienia.....	36
6.3.6.2.	Odczyt składu kolumn.....	37
6.3.6.3.	Odczyt środków służby MPS.....	38
6.3.7.	Zakończenie pracy programu.....	39

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 przesuwać się na setki kilometrów i odległości te pokonywać marszem zachowując jednocześnie całkowitą zdolność do walki.

"M a r s z" jest to zorganizowane przesunięcie wojsk w kolumnach po istniejących drogach i doraźnie przygotowanych drogach na przełaj do wyznaczonego rejonu lub na wyznaczoną rubież. Podczas wykonywania marszu czołgi, artyleria samobieżna i inny sprzęt z małym zapasem przebiegu mogą być przewożone na przyczepach niskopodwoziowych, włączonych w skład kolumn*).

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 sprawności 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 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 (pz) wykonującego marsz.

*) "Regulamin walki wojsk Lądowych SZ RP" część I (dywizja
- pułk) pkt 522 str. 337

W opracowaniu przedstawiono:

- ogólną charakterystykę marszu oraz wynikające z niej właściwości zabezpieczenia w materiały pędne i smary;
- zabezpieczenie materiałowe batalionu i pułku 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 przesunięcie wojsk. Przesunięcia te dokonywane są zarówno w skali operacyjnej, jak i taktycznej.

Przed wybuchem wojny oraz w jej początkowym okresie oraz w czasie jej trwania wojska przesuwa się do rejonu działań bojowych. W rejonie działań bojowych, w celu utworzenia odpowiedniego ugrupowania operacyjnego czy taktycznego dla wykonania określonego zadania bojowego, dokonywane będą przesunięcia oddziałów i pododdziałów, zarówno w kierunku frontu, jak i w stronę skrzydeł, czy też w kierunku odfrontowym.

Przesunięcia oddziałów i pododdziałów mogą być dokonywane na etatowych pojazdach mechanicznych, bez przewożenia ich transportem kolejowym, morskim i powietrznym, oraz sposobem kombinowanym.

Głównym sposobem przesunięcia wojsk jest marsz.

Związek taktyczny (oddział) najczęściej przesuwa się na etatowych pojazdach mechanicznych, co zapewnia jej utrzymanie zwartości oraz gotowości bojowej oddziałów i pododdziałów do wykonania zadań bojowych w dowolnym okresie przesunięcia, zmniejsza wrażliwość na uderzenia przeciwnika, ułatwia manewr, pozwala uzyskać szybkie tempo marszu i zająć planowany rejon w ugrupowaniu dogodnym do wejścia do walki z marszu.

Wojska mogą wykonywać marsz w celu:

- zbliżenia się do przeciwnika z zadaniem rozbicia go;
- dla ześrodkowania w wyznaczonym rejonie;
- opanowania dogodnej rubieży za wycofującym się przeciwnikiem;
- zniszczenia wysadzonego przez przeciwnika desantu.

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 lesistym, bagnistym i innych nieprzyjających warunkach średnia prędkość marszu, w zależności od charakteru terenu i stanu dróg, może się zmniejszyć do 20 km/godz, a długość marszu w ciągu doby wynosić 200-250 km, niekiedy nawet mniej.

Dla odpoczynku żołnierzy, spożycia posiłków, uzupełnienia paliw, sprawdzenia stanu uzbrojenia, sprzętu technicznego, jego obsługi technicznej wyznacza się postoje i odpoczynki.

Postój jednogodzinny wyznacza się po każdym trzech - czterech godzinach marszu oraz jeden postój dwugodzinny w drugiej połowie marszu dobowego. Po wykonaniu każdego marszu dobowego wyznacza się odpoczynki dzienne (nocne).

Od liczby postojów i odpoczynków oraz czasu ich trwania zależą organizacja i praca organów logistycznych w zakresie zabezpieczenia logistycznego, a to z kolei ma wpływ na tworzenie kolumn marszowych pododdziałów (oddziałów) logistycznych i ich miejsca w ugrupowaniu marszowym wojsk.

Powodzenie marszu w znacznym stopniu zależy od terminowego zabezpieczenia logistycznego maszerujących wojsk, a z kolei organizacja tego zabezpieczenia od warunków i sposobów prowadzenia marszu.

Batalion wykonuje marsz w składzie pułku a niekiedy samodzielnie. W składzie pułku może maszerować w składzie sił

głównych lub stanowić awangardę. Batalionowi wyznacza się jedną drogę marszu.

Pułk wykonuje marsz w składzie ZT lub maszeruje samodzielnie. W składzie ZT może wykonywać marsz w kolumnie sił głównych lub stanowić oddział wydzielony. Pułkowi wyznacza się zazwyczaj jedną drogę marszu, a gdy działa w roli oddziału wydzielonego kierunek działania.

Od liczby i stanu otrzymanych do wykorzystania dróg oraz długości kolumn wojsk zależy miejsce poszczególnych pododdziałów (oddziałów) logistycznych w ugrupowaniu marszowym.

Oprócz wymienionych operacyjno-taktycznych warunków prowadzenia marszu na organizację i pracę organów logistycznych pułku 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, a nawet możliwości nieprzyjaciela w zakresie minowania powierzchniowego dróg marszu, działania desantów i grup dywersyjno-rozpoznawczych.

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 wpływ na proces zabezpieczenia wojsk w MPS należy zaliczyć:

- stopień zagrożenia ze strony nieprzyjaciela;
- długość marszu;
- warunki geofizyczne;
- przynależność państwowa terytorium, po którym wykonuje się marsz.

W warunkach braku oddziaływania nieprzyjaciela lub gdy

oddziaływanie to jest niewielkie, marsz będzie wykonywany w sposób płynny, ruch wojsk będzie zgodny z wcześniej opracowanym planem, bez żadnych zatrzymań, konieczności wykonywania objazdów z minimalnymi stratami w ludziach i sprzęcie.

Oddziaływanie nieprzyjaciela broniąmi o różnym zasięgu i możliwościach powoduje utrudnienie, zarówno dla maszerujących wojsk, jak i w procesie ich zaopatrzenia w MPS. Zakłócenia spowodowane tym oddziaływaniem prowadzą do korekt planów przegrupowania, zmieniane są drogi marszu, przeprawy, rejony odpoczynków itp. Zmiany te powodują potrzebę przydzielania innych źródeł zaopatrzenia, ich oddalenie oraz możliwości dystrybucyjne mogą wpłynąć w sposób istotny na czas odtworzenia zapasów w pododdziałach logistycznych maszerujących wojsk.

Oddziaływanie nieprzyjaciela może również spowodować straty w oddziałach (pododdziałach) logistycznych, a także w wydzielonych źródłach zaopatrzenia.

Wpływ takich elementów, jak długość marszu i warunki geograficzne wydają się na tyle oczywiste, że zrezygnowano z ich opisywania w prezentowanej pracy.

Przynależność państwowa terytorium, po którym wykonywany jest marsz ma duży wpływ na możliwości odtworzenia zapasów w pododdziałach (oddziałach) logistycznych maszerujących wojsk. Najlepsze pod tym względem warunki będą miały wojska przegrupowujące się na terytorium kraju. Proces zabezpieczenia wojsk w MPS będzie wówczas realizowany nie tylko siłami i środkami przegrupowujących się wojsk a również siłami i środkami okręgów wojskowych i gospodarki narodowej. Na obszarze państw sojuszniczych wojska nasze otrzymują wsparcie w systemach zaopatrywania i obsługi rozwiniętych na terytorium tych państw. Najtrudniejsze warunki zabezpieczenia maszerujących wojsk w MPS będą panowały na terytoriach państw wrogich, wówczas zabezpieczenie to będzie oparte na polowym systemie zaopatrywania.

2. ZABEZPIECZENIE MATERIAŁOWE BATALIONU I PUŁKU W MARSZU.

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

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

W okresie przygotowania do marszu należy zgromadzić pełne zapasy ruchome paliw na wszystkich szczeblach organizacyjnych w ilości określonej normami oraz zapasy doraźne.

Urzutowanie zapasów ruchomych MPS 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	Razem w ZT
Benzyna samochodowa	1.3	—	—	0.3	0.4	2.0
Olej napędowy:						
- do pojazdów kołowych	1.3	—	—	0.3	0.4	2.0
- do pojazdów gasienicowych i agregatów	1.0	—	—	0.6	0.6	2.2
- do czołgów	1.0	—	0.3	0.6	0.6	2.5

Zapasy doraźne paliw wynoszą:

- benzyny samochodowej i oleju napędowego do pojazdów kołowych - 0.3 jn;
- oleju napędowego do pojazdów gasienicowych 0.2 - 0.5 jn;.

*) Jednostka napełnienia benzyny samochodowej i oleju napędowego do pojazdów kołowych jest to ilość MPS ustalona na jeden pojazd mechaniczny na 500 km przebiegu wg zasadniczej normy zużycia.

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	0.1	0.15	0.3 - 0.32	0.5 - 0.55
PZ	0.1	0.2	0.3 - 0.32	0.5 - 0.55

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 pułku 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 pułk powinien posiadać po wykonaniu marszu*/.

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

- benzyny samochodowej - 0.3jn;
- oleju napędowego do pojazdów kołowych - 0.3jn;
- oleju napędowego do pojazdów gąsienicowych - 0.2 - 0.5jn.

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 nieprzyjaciela 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 - 0.9 - 0.96jn (tj. 17 - 18t);
- oleju napędowego do pojazdów kołowych - 0.9 - 0.96jn (tj. 12.6 - 13.5t);
- oleju napędowego do pojazdów gąsienicowych - 1.5 - 1.6jn (tj. 65 - 80t).

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

**/ Jako marsz dobowy przyjęto odległość 300km.

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 marzu w terenie górzystym i leśnisto-jeziornym - nawet o 50 - 100% normy. Zwiększone potrzeby paliw występują również w warunkach uderzeń nieprzyjaciela 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 pułku uzupełnianie paliw w zbiornikach pojazdów mechanicznych i wozach bojowych dokonywane jest sukcesywnie w czasie postojów i odpoczynków. Jest ono koniecznym zwłaszcza w pojazdach gaśnicowych, 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 pułku 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 zabezpieczonego 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 pułku (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 pułku 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 naruszenia zapasów paliw przewożonych w pododdziałach zaopatrzenia.

Zróżnicami uzupełnienia paliw dla maszerującego pułku w marszu na dużą odległość mogą być: stacjonarne składy MPS wojskowe i cywilne, transporty kolejowe z paliwami na stacjach wyładowania obsługiwane przez grupy tankowania o zdolności wydawczej do 200t paliw na godzinę, 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 napowego napełnienia;
- długość drogi marszu;
- uaktualnioną masę jednostki napełnienia 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 $(1 + k_e) * /$.

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 "kmt"

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

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

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 - 0.6. 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 wyni^osza:

Rodzaj pojazdu	Współczynnik k_{mt}	Współczynnik k_g
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 wprowadzono je do jednego współczynnika syntetycznego oznaczając go jako "K".

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

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:

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

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. WZORY MATEMATYCZNE DO OBLICZANIA PARAMETRÓW OPERACYJNO - LOGISTYCZNYCH W ZAKRESIE MPS

1. Wielkość pojedynczej jednostki napełnienia paliwa do pojazdu mechanicznego wyznacza się wg następującego wzoru:

$$j_n = \frac{S * N}{100} \text{ [dcm}^3 \text{]}$$

gdzie:

- j_n - pojedyncza jednostka napełnienia;
- S - zasięg pojazdu na $1j_n$;
- N - zużycie paliwa na 100km/w litrach.

2. Wielkość zbiorowej jednostki napełnienia paliwa do pojazdów mechanicznych jednego typu wyznacza się wg następującego wzoru:

$$j_n = \frac{S * N}{100} * n \text{ [dcm}^3 \text{]}$$

gdzie:

- n - liczba pojazdów.

3. Wielkość zbiorowej jednostki napełnienia paliwa do pojazdów mechanicznych pododdziału wyznacza się wg następującego wzoru:

$$j_n^r_p = j_n^r_1 + j_n^r_2 + \dots + j_n^r_k$$

gdzie:

- $j_n^r_p$ - jednostka napełnienia p-tego pododdziału r-tym rodzajem paliwa;
- $j_n^r_1$ - jednostka napełnienia i-szego rodzaju pojazdów r-tym rodzajem paliwa;

$$j n_1^r = j n_1^r * n_1$$

gdzie:

$j n_1^r$ - jednostka napełnienia r-tym rodzajem
paliwa i-szego typu pojazdów;

n_1 - liczba pojazdów pierwszego typu
w pododdziale.

k - liczba typów pojazdów mechanicznych w pododdziale
wykorzystujących dany rodzaj paliwa.

6. OPIS PROGRAMU "ZAOPATRZENIE W MPS pz W MARSZU" (MPSMARSZ)

6.1. Przeznaczenie programu

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

6.2. 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_map;
- * 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.

6.3 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 pz WYKONUJĄCEGO MARSZ

opracowany przez:

ppłk 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>.

6.3.1. Zestawienie masy jednostek napełnienia pododdziałów

Program komputerowy MP3MARSZ ma "wazyte" pełne zestawienie jednostek napełnienia pododdziałów pułku zmechanizowanego 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 PUŁKU

Oddział	BS	ONPK	ONPG	ONCZ	suma ON
bcz/T-72/	1.27	0.80	0.50	33.52	34.82
bcz/T-55/	1.27	0.80	0.50	26.80	28.10
bpz	2.46	1.52	0.46	-----	1.98
bpzmot	2.46	1.52	0.46	-----	1.98
da	2.60	1.08	0.47	-----	1.55
das	1.85	1.36	6.67	-----	8.03
daplot	2.05	2.86	2.05	-----	4.91
dappanc	1.81	-----	-----	-----	-----
kr	1.30	-----	-----	-----	-----
ksap	1.04	1.39	4.22	-----	5.61
kl	0.50	1.50	-----	-----	1.50
kmed	1.04	-----	-----	-----	-----
kzaop	6.28	6.56	-----	-----	6.56
krem	8.35	1.66	2.66	-----	4.32
plochm	0.72	0.39	-----	-----	0.39
ploirr	0.49	0.26	-----	-----	0.26
SD	0.25	0.15	-----	-----	0.15
CKWL	0.30	-----	-----	-----	-----

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>.

6.3.2. 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>.

6.3.2.1. 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 PODODZIAŁÓW

bcz/T-72/	bcz/T-55/	bpz	bpzmot	da	das	daplot
bappanc	kr	ksap	kl	kmed	kzaop	krem
plochem	ploirr	SD	CKWL			

Kolumna nr: 1

Wprowadź kolejną nazwę pododziału:

Użytkownik wprowadza nazwę pododziału ze zbioru DOPUSZCZALNYCH NAZW PODODZIAŁÓW, 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/	bpz	bpzmot	da	das	daplot
bappanc	kr	ksap	kl	kmed	kzaop	krem
plochem	ploirr	SD	CKWL			

Kolumna nr: 1

Wprowadź kolejną nazwę pododdziału:

Podaj procent ukończenia:

1/2 bcz/T-72/	1/2 bpzmot	1/6 ksap
1/2 das	CKWL	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.

6.3.2.2. 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).

Ponaciśnięciu klawisza <ENTER> na ekranie monitora pojawi się skład organizacyjny zadanej kolumny.

Przykład:

SKŁAD KOLUMNY NR 1		
1/2 bcz/T-72/	1/2 bpzmot	1/6 ksap
1/2 das	CKWL	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.

6.3.3. Obliczanie masy jednostek napełnienia kolumn marszowych.

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

ZESTAWIENIE MASY JEDNOSTEK NAPEŁNIENIA KOLUMN

kolumny	BS	ONPK	ONPG	ONCZ	Suma ON
1	6.33	4.98	4.33	15.08	24.40
2	6.69	5.66	2.25	15.08	23.00

Naciśnij ENTER

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

6.3.4. 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
KONCZ = 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:

ZUZYCIE MPS NA DRODZE DO REJONU DWUGODZINNEGO POSTOJU

Kolumna	ZuBS	ZuONPK	ZuONPG	ZuONCZ	sumaON	
1	2.66	2.09	1.95	9.05	13.09	KBS = 0.28
2	3.16	2.63	2.36	9.05	14.05	KONPK = 0.28

suma	5.82	4.73	4.31	18.10	27.14	KONPG = 0.30 KONCZ = 0.40

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 ODPOCZYNKU

Kolumna	ZuBS	ZuONPK	ZuONPG	ZuONCZ	sumaON	
1	3.01	2.37	2.21	10.26	14.84	KBS = 0.28
2	3.58	2.98	2.68	10.26	15.92	KONPK = 0.28

suma	6.59	5.36	9.20	38.62	53.17	KONPG = 0.30 KONCZ = 0.40

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

Naciśnij <ENTER>

ZUZYCIE MPS W CZASIE CAŁEGO MARSZU

Kolumna	ZuBS	ZuONPK	ZuONPG	ZuONCZ	sumaON	
1	5.67	4.46	4.16	19.31	27.93	KBS = 0.28
2	6.74	5.62	5.04	19.31	29.97	KONPK = 0.28

suma	12.41	10.08	18.41	77.23	105.72	KONPG = 0.30 KONCZ = 0.40

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.

6.3.5. 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:

6.3.5.1. 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 środ-

kach służby MPS ekranu 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 z przyczepa	CD7.5+CP11	0	1

ilość BS 8.2 t

ilość ON 25.6 t

6.3.5.2. 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:

SIŁY 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 z przyczepa	CD7.5+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>.

6.3.5.3. Podział sił i środków służby MPS

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

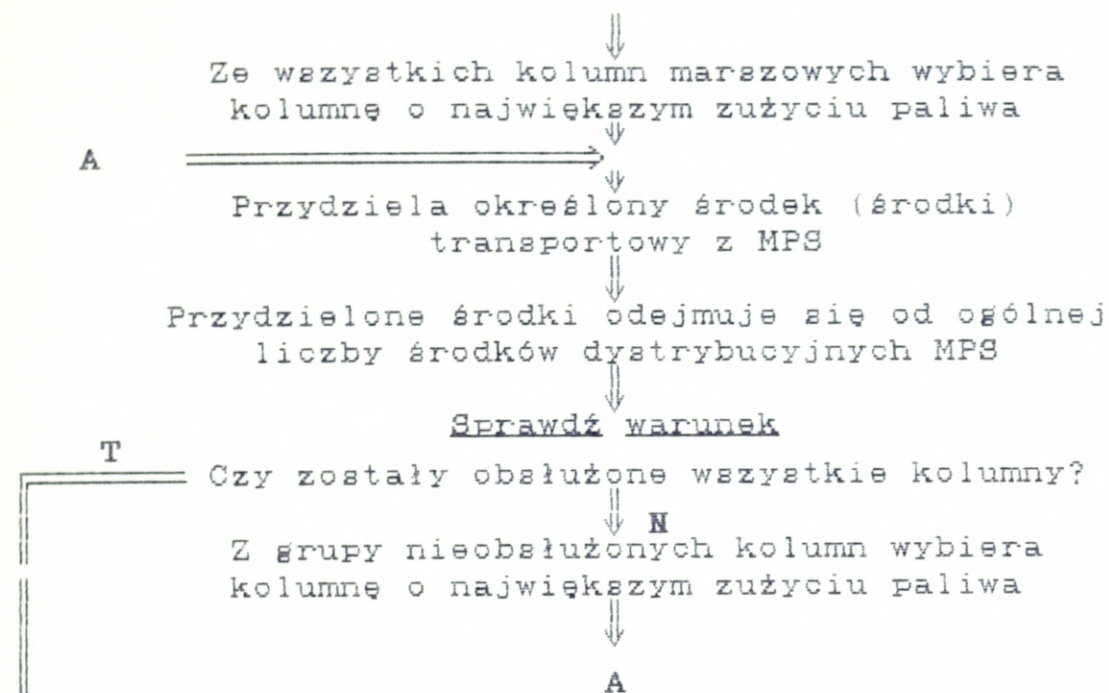
Podaj priorytet przydziału paliwa dla czołgów (1-3)

Autorzy programu uważali za celowe wprowadzenie takiej modyfikacji, ponieważ czołgi z jednej strony stanowią największy potencjał bojowy pułku, 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.

Program działa w następujący sposób:



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 CKWL (w kolumnach tych będzie najwięcej samochodów).

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

PODZIAŁU SIŁ I ŚRODKÓW SŁUŻBY MPS

kol.	STRB		CD4.5		CD5W		CDPO4W		CD7.5		STRK		SD5WK		CP		PT		CD7.5+CP11	
	bs	on	bs	on	bs	on	bs	on	bs	on	bs	on	bs	on	bs	on	bs	ob	bs	on
1	--	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 ZUŻYCIA PALIW

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

Naciśnij <ENTER>

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

6.3.6. Operacje dyskowe

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

O P E R A C J E D Y S K O W E

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.

6.3.6.1. 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	ONCZ
1	1.27	0.80	0.50	33.52
2	1.27	0.80	0.50	26.80
3	2.46	1.52	0.46	-----
4	2.46	1.52	0.46	-----
5	2.60	1.08	0.47	-----
6	1.85	1.36	6.67	-----
7	2.05	2.86	2.05	-----
8	1.81	-----	-----	-----
9	1.30	-----	-----	-----
10	1.04	1.39	4.22	-----
11	0.50	1.50	-----	-----
12	1.04	-----	-----	-----
13	6.28	6.56	-----	-----
14	8.35	1.66	2.66	-----
15	0.72	0.39	-----	-----
16	0.49	0.26	-----	-----
17	0.25	0.15	-----	-----
18	0.30	-----	-----	-----

Naciśnij <ENTER>

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

W przypadku, gdy zbiór o nazwie JED_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.

6.3.6.2. Odczyt składu kolumn

Wybierając te opcje 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	SD	1/2kzaop

Naciśnij <ENTER>

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

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

Brak zbioru KOLUMNY

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.

6.3.6.3. 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 SŁUŻBY MPS

środek transportu	ilość sztuk	BS	ON
samochód 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
samochód transportowy z karnistr.	STRK	---	---
cysterna dystrybutor	CD5WK	---	---
cysterna przyczepa	CP	1	1
przyczepa transportowa z beczkami	PT	---	2
cysterna z przyczepa	CD7.5+CP11	1	2

Naciśnij <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

Naciśnij <ENTER>

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

6.3.7. Zakończenie pracy programu

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

Wydrukowano w 3 egz.
Egz.Nr 1-3 - Bibl.AON
Wyk. ppłk KLIMKIEWICZ tel.13-703
Druk JB dn.3.04.93 r.
Nr masz. PF2/AON

Załącznik nr 1.

Postać źródłowa programu

```
1
2 Program mpsmars2;
3 uses Crt,DOS,printer;
4 Const
5     ilosc_podod = 18;
6     ilosc_kol   = 10;
7     ilosc_sr    = 10;
8     benzyna     = 1;
9     olej        = 2;
10    STRB        = 1;
11    CD45        = 2;
12    CD5W        = 3;
13    CDPO4W     = 4;
14    CD75        = 5;
15    STRK        = 6;
16    CD5WK       = 7;
17    CP          = 8;
18    PT          = 9;
19    CD75CP11   = 10;
20    SD          = 17;
21    CKWL       = 18;
22
23 Type
24     jedn_nap   = record
25         BS     : Real;
26         ONPK   : Real;
27         ONPG   : Real;
28         ONCZ   : Real;
29     end;
30     opis_kol   = record
31         bs,on  : byte;
32     end;
33     opis_zap   = record
34         ben    : real;
35         pbs    : real;
36         oncz   : real;
37         poncz  : real;
38         onkg   : real;
39         ponkg  : real;
40     end;
41     opis_przydz = record
42         ben    : integer;
43         on     : integer;
44     end;
45     skl_kol    = record
46         nr_odd : Integer;
47         wlk_odd : string[3];
48         ulamek : Real;
49         proc   : Real;
50         strt   : Real;
51     end;
52     opis_cyst  = record
53         naz    : String[40];
54         ben    : Integer;
55         on     : Integer;
```

```
56     zbn    : Integer;
57     zon    : Integer;
58     pojem : Real;
59     end;
60     Nazwa = String[14];
61
62 var
63     wyj : text;
64     kolejnosc : array[1..ilosc_kol] of opis_kol;
65     przydzial_BS, przydzial_ON : real;
66     tablica_przydzialu : Array[1..ilosc_kol, 1..ilosc_sr] of opis_przydz;
67     zap_kol : Array[1..ilosc_kol] of opis_zap;
68     cysterny : Array[1..ilosc_sr] of opis_cyst;
69     tab_jed_nap : Array[1..ilosc_podod] of jedn_nap;
70     nazwa_podod : Array[1..ilosc_podod] of Nazwa;
71     kolumny : Array[1..ilosc_kol, 1..ilosc_podod] of skl_kol;
72     polecenie_da : Char;
73     k, j, numer_odd, rejs : Integer;
74     nazwa_odd : Nazwa;
75     numer_kol : Integer;
76     kbd : char;
77     numer : String[4];
78     kreski : Byte;
79     wiersz : String[70];
80     nie_wym : Boolean;
81     wykonano : Boolean;
82     MBS, MON, MONPK, MONPG, MONCZ, ul : Real;
83     KBS, KONPK, KONPG, KONCZ : Real;
84     ZuBS, ZuONPK, ZuONPG, ZuONCZ, ZuON, suma : Real;
85     cyfra_1, cyfra_2, nr_linii : Integer;
86     suma_ben, suma_on : Real;
87     ilosc_ben, ilosc_on : Integer;
88     pomijac : Boolean;
89     L1, L12, L2 : Real;
90     sa_cysterny : Boolean;
91     przydzielono, czolgi, jest : Boolean;
92     paliwo, cyst : integer;
93     BufLen : integer;
94     priorytet : real;
95     nr_klm : integer;
96
97
98 function mamy(cyster : integer) : Boolean;
99 begin
100     mamy:=false;
101     with cysterny[cyster] do
102         begin
103             case paliwo of
104                 benzyna : if ((ben-zbn)>0) then mamy:=true;
105                 olej : if ((on -zon)>0) then mamy:=true;
106             end;
107         end;
108     end;
109
110 function potrzebują_czolgi : boolean;
```

```
111 begin
112   potrzebuja_czolgi:=false;
113   for k:=1 to ilosc_kol do
114     begin
115       with zap_kol[k] do
116         begin
117           if ((oncz-poncz) > 0) then
118             begin
119               potrzebuja_czolgi:=true;
120               exit;
121             end;
122         end;
123     end;
124 end;
125
126 Procedure szuk_max_czolgi;
127 var
128   zap_on : real;
129 begin
130   zap_on:=0;
131   jest:=false;
132   for k:=1 to ilosc_kol do
133     begin
134       with zap_kol[k] do
135         begin
136           if ((oncz-poncz) > zap_on) then
137             begin
138               zap_on:=oncz-poncz;
139               jest:=true;
140               nr_klm:=k;
141             end;
142         end;
143     end;
144 end;
145
146 procedure szuk_kol(n_podod : integer);
147 begin
148   nr_klm:=0;
149   for k:=1 to ilosc_kol do
150     begin
151       if kolumny[k,1].nr_odd<>0 then
152         begin
153           for j:=1 to ilosc_podod do
154             begin
155               if kolumny[k,j].nr_odd=n_podod then
156                 begin
157                   nr_klm:=k;
158                   exit;
159                 end;
160             end;
161           end;
162         end;
163     end;
164
165 Procedure szuk_max_zap(wlk_zap : integer);
```

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

```
221 with zap_kol[kol_num] do
222   begin
223     if (paliwo=olej) then
224       begin
225         przydzial_ON:=przydzial_ON+cysterny[cyst_rodz].pojem*0.82;
226         if czolgi then
227           poncz:=poncz+cysterny[cyst_rodz].pojem*0.82
228         else
229           ponkg:=ponkg+cysterny[cyst_rodz].pojem*0.82;
230       end
231     else
232       begin
233         pbs:=pbs+cysterny[cyst_rodz].pojem*0.78;
234         przydzial_BS:=przydzial_BS+cysterny[cyst_rodz].pojem*0.78;
235       end;
236     end;
237   with cysterny[cyst_rodz] do
238     begin
239       case paliwo of
240         benzyna : zbn:=zbn+1;
241         olej    : zon:=zon+1;
242       end;
243     end;
244   end;
245
246 Procedure przydz(kol_num,cyst_rodz: integer);
247   var
248     pojazdy : boolean;
249   begin
250     if ((cyst_rodz=STRB) or (cyst_rodz=CDPO4W) or (cyst_rodz=PT)) and czolgi then
251       przydzial(kol_num,cyst_rodz)
252     else
253       if ((cyst_rodz=STRB) or (cyst_rodz=CDPO4W) or (cyst_rodz=PT)) and not czolgi then
254         begin
255           if potrzebuja_czolgi then
256             begin
257               szuk_max_czolgi;
258               if jest then
259                 begin
260                   pojazdy:=czolgi;
261                   czolgi:=true;
262                   przydzial(kol_num,cyst_rodz);
263                   czolgi:=pojazdy;
264                 end;
265             end
266           else przydzial(kol_num,cyst_rodz);
267         end
268       else przydzial(kol_num,cyst_rodz);
269     end;
270
271 Procedure szuk_max_cyst;
272   var
273     wlk_cyst, wlk : real;
274   begin
275     jest:=false;
```

```
276   wlk_cyst:=0;
277   for k:=1 to ilosc_sr do
278   begin
279     with cysterny[k] do
280     begin
281       wlk:=0;
282       case paliwo of
283         benzyna : if ((ben-zbn)>0) then wlk:=pojem;
284         olej     : if ((on -zon)>0) then wlk:=pojem;
285       end;
286       if (wlc_cyst<wlc) then
287       begin
288         jest:=true;
289         wlk_cyst:=wlc;
290         cyst:=k;
291       end;
292     end;
293   end;
294 end;
295
296 Procedure rozdzial(cst : integer);
297 var
298   kolumn : integer;
299 begin
300   jest:=true;
301   while (mamy(cst) and jest) do
302   begin
303     przydzielono:=false;
304     szuk_max_zap(0);
305     if jest then
306     begin
307       kolumn:=nr_klm;
308       szuk_kol(SD);
309       if (kolumn=nr_klm) then
310       begin
311         if mamy(STRK) then przydzial(nr_klm,STRK);
312         przydzielono:=true;
313       end
314       else
315       begin
316         szuk_kol(CKWL);
317         if (kolumn=nr_klm) then
318         begin
319           if mamy(STRK) then przydzial(nr_klm,STRK);
320           przydzielono:=true;
321         end;
322       end;
323       if not przydzielono then
324         przydzial(kolumn,cst);
325     end;
326   end;
327 end;
328
329 Procedure rozdzielenie;
330 begin
```

```
331   rozdzial(CD75CP11);
332   rozdzial(CP);
333   rozdzial(CDP04W);
334   rozdzial(STRB);
335   rozdzial(CD5WK);
336   rozdzial(CD45);
337   rozdzial(CD5W);
338   rozdzial(CD75);
339   rozdzial(STRK);
340   rozdzial(PT);
341   end;
342
343 Procedure czekanie_na_return;
344   var
345     znak : Char;
346
347   begin
348     GoToxy(30,24);
349     Write('Nacisnij ENTER ');
350     repeat
351       znak:=readkey;
352     until znak = chr(13);
353   end;
354
355 Procedure wysw_rozdzialu;
356   begin
357     ClrScr;
358     gotoxy(1,1);
359     Writeln('                PODZIAL SIL I SRODKOW SLUZBY MPS                ');
360     writeln;
361     NormVideo;
362     Writeln('          STRB  CD4.5  CD5W  CDP04W  CD7.5  STRK  CD5WK  CP    PT  CD7.5+CP11');
363     writeln;
364     Writeln('kolum  bs on  bs on  bs on  bs on  bs on  bs on  bs on  bs on  bs on  ');
365     writeln;
366     for k:=1 to ilosc_kol do
367       begin
368         write(' ',k,' ');
369         for j:=1 to ilosc_sr do
370           begin
371             if (tablica_przydzialu[k,j].ben=0) then write(' --')
372             else write(tablica_przydzialu[k,j].ben:4);
373             if (tablica_przydzialu[k,j].on =0) then write(' --')
374             else write(tablica_przydzialu[k,j].on:3);
375           end;
376         writeln;
377       end;
378   end;
379
380 Procedure ilosc_benzyny(n_r : integer);
381   var
382     l : integer;
383   begin
384     suma_ben:=0;
385     suma_on:=0;
```

```
386     for l:=1 to n_r do
387     begin
388         suma_ben:=suma_ben+cysterny[l].pojem*cysterny[l].ben*0.78;
389         suma_on :=suma_on +cysterny[l].pojem*cysterny[l].on *0.82;
390     end;
391 end;
392
393 Procedure obl_jed_nap_kol(kol : Integer);
394 var
395     akt_oddzial, nr_od : Integer;
396     ulam,pr,strat      : Real;
397 begin
398     akt_oddzial:=1;
399     MBS :=0;
400     MONPK:=0;
401     MONPG:=0;
402     MONCZ:=0;
403     if (kolumny[kol,1].nr_odd>0) then
404     begin
405         while (kolumny[kol,akt_oddzial].nr_odd > 0) do
406         begin
407             nr_od:=kolumny[kol,akt_oddzial].nr_odd;
408             ulam :=kolumny[kol,akt_oddzial].ulam;
409             pr  :=kolumny[kol,akt_oddzial].proc/100.0;
410             with tab_jed_nap[nr_od] do
411             begin
412                 MBS :=MBS +BS *ulam*pr;
413                 MONPK:=MONPK+ONPK*ulam*pr;
414                 MONPG:=MONPG+ONPG*ulam*pr;
415                 MONCZ:=MONCZ+ONCZ*ulam*pr;
416             end;
417             akt_oddzial:=akt_oddzial+1;
418         end;
419     end;
420 end;
421
422 Procedure podzial;
423 var
424     jnBS,jnONPK,jnONPG,jnONCZ : Real;
425     WBS,WON,WONPK,WONPG,WONCZ,UBS,UGN,pMBS,pMON : Real;
426
427 begin
428     for k:=1 to ilosc_sr do
429     begin
430         cysterny[k].zbn:=0;
431         cysterny[k].zon:=0;
432     end;
433     if (L12 = 0) then
434     begin
435         gotoxy(20,18);
436         WriteLn('Nie podano parametrow marszu',chr(7));
437         czekanie_na_return;
438         wykonano:=false;
439     end
440     else
```

```
441 if not sa_cysterny then
442   begin
443     gotoxy(15,18);
444     WriteLn('Nie przydzielono srodkow sluzby mps',chr(7));
445     czekanie_na_return;
446     wykonano:=false;
447   end
448 else
449   begin
450     clrscr;
451     przydzial_BS:=0;
452     przydzial_ON:=0;
453     gotoxy(5,5);
454     write(' Podaj priorytet przydzialu paliwa dla czolgow (1-3) ');
455     gotoxy(58,5);
456     read(priorytet);
457     Clrscr;
458     for k:=1 to ilosc_kol do
459       begin
460         obl_jed_nap_kol(k);
461         with zap_kol[k] do
462           begin
463             ben:=KBS*L12*MBS/100;
464             pbs:=0;
465             oncz:=KONCZ*L12*MONCZ/100;
466             poncz:=0;
467             onkg:=(KONPG*MONPG+KONPK*MONPK)*L12/100;
468             ponkg:=0;
469           end;
470         end;
471     for k:=1 to ilosc_kol do
472       for j:=1 to ilosc_sr do
473         begin
474           tablica_przydzialu[k,j].ben:=0;
475           tablica_przydzialu[k,j].on :=0;
476         end;
477     paliwo:=benzyna;
478     czolgi:=false;
479     if mamy(STRK) then
480       begin
481         szuk_kol(SD);
482         if (nr_klm>0) then przydzial(nr_klm,STRK);
483       end;
484     if mamy(STRK) then
485       begin
486         szuk_kol(CKWL);
487         if (nr_klm>0) then przydzial(nr_klm,STRK);
488       end;
489     paliwo:=olej;
490     if mamy(STRK) then
491       begin
492         szuk_kol(SD);
493         if (nr_klm>0) then przydzial(nr_klm,STRK);
494       end;
495     if mamy(STRK) then
```

```
496         begin
497             szuk_kol(CKWL);
498             if (nr_klm>0) then przydzial(nr_klm,STRK);
499         end;
500     end;
501     czolgi:=false;
502     paliwo:=benzyna;
503     repeat
504         szuk_max_zap(0);
505         if jest then szuk_max_cyst;
506         if jest then przydzial(nr_klm,cyst);
507     until not jest;
508     czolgi:=true;
509     paliwo:=olej;
510     repeat
511         if mamy(STRB) then szuk_max_czolg;
512         if jest then przydzial(nr_klm,STRB);
513     until not (jest and mamy(STRB));
514     repeat
515         jest:=false;
516         if mamy(PT) then
517             szuk_max_czolg;
518         if jest then przydzial(nr_klm,PT);
519     until not(jest and mamy(PT));
520     repeat
521         jest:=false;
522         if mamy(CDPO4W) then szuk_max_czolg;
523         if jest then przydzial(nr_klm,CDPO4W);
524     until not (jest and mamy(CDPO4W));
525     paliwo:=olej;
526     repeat
527         szuk_max_zap(0);
528         if jest then
529             szuk_max_cyst;
530         if jest then
531             przydzial(nr_klm,cyst);
532     until not jest;
533     paliwo:=olej;
534     rozdzielanie;
535     paliwo:=benzyna;
536     rozdzielanie;
537     wysw_rozdzialu;
538     czekanie_na_return;
539
540     {*****}
541     {#                                     #}
542     {# Obliczanie calkowitej masy napelnienia pulku #}
543     {#                                     #}
544     {*****}
545
546     jnBS :=0;
547     jnDNPK:=0;
548     jnDNPG:=0;
549     jnONCZ:=0;
550     for k:=1 to ilosc_kol do
```

```

551     begin
552         obl_jed_nap_kol(k);
553         jnBS :=jnBS +MBS;
554         jnONPK:=jnONPK+MONPK;
555         jnONPG:=jnONPG+MONPG;
556         jnONCZ:=jnONCZ+MONCZ;
557     end;
558
559     {*****}
560     {*                                     *}
561     {* Obliczanie ilosci paliwa jaka pozostaa przy sprzecie *}
562     {*                                     *}
563     {*****}
564
565     MBS :=jnBS *1.3-jnBS *KBS *L12/100;
566     MONPK:=jnONPK*1.3-jnONPK*KONPK*L12/100;
567     MONPG:=jnONPG -jnONPG*KONPG*L12/100;
568     MONCZ:=jnONCZ -jnONCZ*KONCZ*L12/100;
569
570     {*****}
571     {*                                     *}
572     {* Przejscie na rope i benzyne *}
573     {*                                     *}
574     {*****}
575
576     pMBS:=MBS;
577     MON :=MONPK+MONPG+MONCZ;
578     pMON:=MON;
579
580     {*****}
581     {*                                     *}
582     {* W cysternach mamy suma_on i suma_ben *}
583     {*                                     *}
584     {*****}
585
586     pMBS:=MBS+przydzial_BS;           { stan benzyny }
587     pMON:=MON+przydzial_ON;           { stan rope   }
588     ClrScr;
589     LowVideo;
590     gotoxy(21,10);
591     writeln(' BILANS ZUZYCIA PALIW ');
592     writeln;
593     NormVideo;
594     writeln;
595     writeln('          BS          ON');
596     writeln;
597     writeln('W rejonie wyjsciowym pozostalo :          ',pMBS:6:2,'          ',pMON:6:2);
598     writeln;
599     MBS:=jnBS*1.3-pMBS;
600     MON:=jnONPK*1.3+jnONPG+jnONCZ-pMON;
601     writeln('DO PELNYCH ZAPASOW RUCHOMYCH:');
602     writeln;
603     writeln('Przy sprzecie nalezy uzupelnic :          ',MBS:6:2,'          ',MON:6:2);
604     writeln;
605     MBS:=jnBS*1.6-pMBS;

```

```
606     MON:=jnDNPK*1.6+jnDNPG*1.6+1.9*jnONCZ-pMON;
607     writeln('W pulku nalezy uzupelnic :           ',MBS:6:2,'           ',MON:6:2);
608     czekanie_na_return;
609 end;
610
611 Function pobranie_numeru(xpos,ypos : Integer) : Integer;
612 var
613     nr_kol : Integer;
614 begin
615     repeat
616         nr_kol:=0;
617         GoToxy(xpos,ypos);
618         ClrEol;
619         ReadLn( numer );
620         val( numer, nr_kol, k );
621     until k=0;
622     pobranie_numeru:=nr_kol;
623 end;
624
625 Procedure wpr_danych;
626 begin
627     wykonano:=true;
628     ClrScr;
629     gotoxy(10,6);
630     LowVideo;
631     WriteLn(' WPROWADZENIE DANYCH O SILACH I SRODKACH SLUZBY MPS ');
632     NormVideo;
633     WriteLn;
634     WriteLn(' srodek transportu           ilosc szt:           BS           ON');
635     WriteLn;
636     nr_linii:=10;
637     for j:=1 to ilosc_sr do
638     begin
639         gotoxy(1,nr_linii);
640         Write(cysterny[j].naz);
641         ilosc_ben:=pobranie_numeru(50,nr_linii);
642         ilosc_on :=pobranie_numeru(60,nr_linii);
643         cysterny[j].ben:=ilosc_ben;
644         cysterny[j].on :=ilosc_on;
645         cysterny[j].zbn:=0;
646         cysterny[j].zon:=0;
647         ilosc_benzyny(j);
648         nr_linii:=nr_linii+1;
649         gotoxy(10,22);
650         Write('ilosc BS ',suma_ben:6:1,' t           ilosc ON ',suma_on:6:1,' t');
651     end;
652     sa_cysterny:=true;
653 end;
654
655 Procedure wysw_danych;
656 begin
657     if sa_cysterny then
658     begin
659         ClrScr;
660         gotoxy(20,1);
```

```
661 LowVideo;
662 WriteLn(' SILY I SRODKI SLUZBY MPS ');
663 NormVideo;
664 WriteLn;
665 WriteLn(' srodek transportu      ilosc :      BS      ON');
666 nr_linii:=4;
667 for k:=1 to ilosc_sr do
668   begin
669     gotoxy(1,nr_linii);
670     Write(cysterny[k].naz);
671     gotoxy(49,nr_linii);
672     if cysterny[k].ben > 0 then
673       Write(cysterny[k].ben:3)
674     else
675       Write('---');
676     gotoxy(59,nr_linii);
677     if cysterny[k].on > 0 then
678       WriteLn(cysterny[k].on:3)
679     else
680       WriteLn('---');
681     nr_linii:=nr_linii+2;
682   end;
683 WriteLn;
684 writeln;
685 czekanie_na_return;
686 if L1 > 0 then
687   begin
688     ZuBS:=0;
689     ZuON:=0;
690     for k:=1 to ilosc_kol do
691       begin
692         obl_jed_nap_kol(k);
693         ZuBS:=ZuBS+MBS*L12*KBS/100;
694         MONPK:=MONPK*KONPK+MONPG*KONPG+MONCZ*KONCZ;
695         ZuON:=ZuON+MONPK*L12/100;
696       end;
697     Write(' ');
698     ClrScr;
699     gotoxy(15,5);
700     LowVideo;
701     WriteLn(' POROWNANIE POTRZEB Z MOZLIWOSCAMI ');
702     NormVideo;
703     writeln;
704     WriteLn('          Zuzycie          Przydzial');
705     writeln;
706     WriteLn('          BS          ON          BS          ON');
707     writeln;
708     gotoxy(6,11);
709     Write(ZuBS:6:2,' t');
710     gotoxy(20,11);
711     Write(ZuON:6:2,' t');
712     gotoxy(40,11);
713     Write(suma_ben:6:2,' t');
714     gotoxy(52,11);
715     WriteLn(suma_on:6:2,' t');
```

```
716     if ((ZuBS > 0) or (ZuON > 0)) then
717         begin
718             gotoxy(15,16);
719             WriteLn('      POSIADANA ILOSC PALIW STANOWI');
720             writeln;
721             WriteLn('          BS          ON');
722             gotoxy(20,20);
723             if (ZuBS > 0) then
724                 begin
725                     if (ZuBS>0) then
726                         ZuBS:=suma_ben*100/ZuBS
727                     else
728                         ZuBS:=0;
729                     Write(ZuBS:6:1,' % zuzycia');
730                 end
731             else Write('----');
732             gotoxy(39,20);
733             if (ZuON > 0) then
734                 begin
735                     if (ZuON>0) then
736                         ZuON:=suma_on*100/ZuON
737                     else
738                         ZuON:=0;
739                     Write(ZuON:6:1,' % zuzycia');
740                 end
741             else Write('----');
742         end;
743     end;
744 end
745 else
746     begin
747         wykonano:=false;
748         gotoxy(15,18);
749         WriteLn('Nie przydzielono srodkow sluzby mps',chr(7));
750     end;
751     czekanie_na_return;
752 end;
753
754 Procedure podzial_srodkow;
755 begin
756     wykonano:=true;
757     { cysterny }
758     repeat
759         if wykonano then
760             begin
761                 ClrScr;
762                 GoToxy(12,2);
763                 LowVideo;
764                 WriteLn('      ANALIZA I PODZIAL POSIADANYCH SIL I SRODKOW SLUZBY MPS ');
765                 NormVideo;
766                 WriteLn;
767                 WriteLn('          1. Wprowadzanie danych o silach i srodkach sluzby mps');
768                 writeln;
769                 WriteLn('          2. Wswietlenie danych ');
770                 writeln;
```

```
771     WriteLn('          3. Podzial sil i srodkow sluzby mps');
772     writeln;
773     WriteLn('          4. Powrot do glownego menu');
774     WriteLn;
775     writeln;
776     Write('          polecenie - ');
777 end
778 else
779     begin
780         for k:=10 to 24 do
781             begin
782                 GoToxy(1,k);
783                 ClrEol;
784             end;
785         end;
786         gotoxy(43,13);
787         write(' ');
788         GoToxy(43,13);
789         repeat
790             Read(polecenie);
791             until ((polecenie > '0') and (polecenie < '5'));
792             wykonano:=true;
793             case polecenie of
794                 '1' : wpr_danych;
795                 '2' : wysw_danych;
796                 '3' : podzial;
797             end;
798             until polecenie='4';
799         end;
800
801 Procedure wczytaj_liczba(xpos,ypos,dl_licz : Integer ;dgran,gggran : Real; var liczba : Real);
802 begin
803     repeat
804         GoToxy(xpos,ypos);
805         ClrEol;
806         Readln(liczba);
807         val(liczba,liczba,k);
808         until ((k = 0) and (liczba >= dgran) and (liczba <= gggran));
809     end;
810
811 Function jest_kolumna : Boolean;
812 begin
813     jest_kolumna:=False;
814     for k:=1 to ilosc_kol do
815         if kolumny[k,1].nr_odd > 0 then
816             begin
817                 jest_kolumna:=True;
818                 exit;
819             end;
820     end;
821
822 Procedure obliczenie_zuzycia;
823 var
824     droga,sON,sBS,sONPK,sDNPG,sONCZ :Real;
825     byla_kol : Boolean;
```

```
826     nr_linii : Integer;
827
828 begin
829     if jest_kolumna then
830         begin
831             repeat
832                 ClrScr;
833                 GoToxy(15,1);
834                 LowVideo;
835                 WriteLn(' OBLICZANIE ZUZYCIA MPS PRZEZ POSZCZEGOLNE KOLUMNY');
836                 NormVideo;
837                 WriteLn;
838                 WriteLn(' Zuzycie bedzie obliczane wedlug nastepujacego wzoru :');
839                 WriteLn;
840                 WriteLn('           L * jn');
841                 WriteLn('       Zu = K -----');
842                 WriteLn('           100');
843                 WriteLn('Oznaczenia :');
844                 writeln('-----');
845                 WriteLn(' Zu - zuzycie mps w tonach');
846                 writeln;
847                 WriteLn(' L - wielkosc przebytej drogi w marszu');
848                 writeln;
849                 WriteLn(' jn - wielkosc jednostki napelnienia kolumny marszowej w tonach');
850                 writeln;
851                 WriteLn(' K - wspolczynnik zuzycia mps');
852                 czekanie_na_return;
853                 ClrScr;
854                 gotoxy(21,3);
855                 write(' P A R A M E T R Y   W E J S C I D W E ');
856                 gotoxy(20,4);
857                 write('-----');
858                 gotoxy(9,7);
859                 WriteLn('Podaj wartosci wspolczynnikow K dla poszczegolnych rodzajow paliw');
860                 writeln;
861                 gotoxy(9,9);
862                 write('     KBS = ');
863                 wczytaj_liczbe(20,9,4,0,1,KBS);
864                 gotoxy(9,10);
865                 write('     KONPK = ');
866                 wczytaj_liczbe(20,10,4,0,1,KONPK);
867                 gotoxy(9,11);
868                 write('     KONPG = ');
869                 wczytaj_liczbe(20,11,4,0,1,KONPG);
870                 gotoxy(9,12);
871                 write('     KONCZ = ');
872                 wczytaj_liczbe(20,12,4,0,1,KONCZ);
873                 writeln;
874                 gotoxy(9,15);
875                 write('     M A R S Z ');
876                 gotoxy(9,17);
877                 write('Podaj dlugosc marszu do rejonu dwugodzinnego postoju L1 = ');
878                 wczytaj_liczbe(67,17,6,0,400,L1);
879                 gotoxy(9,18);
880                 write('Podaj dlugosc calego marszu           L12 = ');
```

```
881 wczytaj_liczbe(67,18,6,L1,400,L12);
882 czekanie_na_return;
883 repeat
884   clrscr;
885   gotoxy(20,2);
886   LowVideo;
887   writeln(' OBLICZANIE ZUZYCIA MPS W CZASIE MARSZU');
888   NormVideo;
889   writeln;
890   writeln(' 1. Do rejonu dwugodzinnego postoju');
891   writeln;
892   writeln(' 2. Od rejonu dwugodzinnego postoju do rejonu dziennego/nocnego/ odpoczynku');
893   writeln;
894   writeln(' 3. W czasie calego marszu');
895   writeln;
896   writeln(' 4. Zadanie innych parametrow marszu');
897   writeln;
898   writeln(' 5. Powrot do glownego menu');
899   writeln;
900   writeln;
901   writeln('          Polecenie - ');
902   gotoxy(31,15);
903   write(' ');
904   gotoxy(31,15);
905   repeat
906     read(polecenie);
907   until ((polecenie > '0') and (polecenie < '6'));
908   if polecenie < '4' then
909     begin
910       sON:=0;
911       sBS:=0;
912       sONPK:=0;
913       nr_linii:=4;
914       byla_kol:=false;
915       case polecenie of
916         '1' : droga:=L1;
917         '2' : droga:=L12 - L1;
918         '3' : droga:=L12;
919       end;
920       for k:=1 to ilosc_kol do
921         begin
922           if kolumny[k,1].nr_odd > 0 then
923             begin
924               obl_jed_nap_kol(k);
925               ZuBS :=KBS *droga *MBS/100;
926               ZuONPK:=KONPK*droga*MONPK/100;
927               ZuONP6:=KONP6*droga*MONP6/100;
928               ZuONCZ:=KONCZ*droga*MONCZ/100;
929               sBS:=sBS+ZuBS;
930               sONPK:=sONPK+ZuONPK;
931               sONP6:=sONP6+ZuONP6;
932               sONCZ:=sONCZ+ZuONCZ;
933               byla_kol:=true;
934               nr_linii:=nr_linii+1;
935               if nr_linii = 5 then
```

```
936     begin
937         clrscr;
938         gotoxy(1,1);
939         case polecenie of
940             '1' : begin
941                 writeln('          ZUZYCIE MPS NA DRODZE DO REJONU DWUGODZINNEGO POSTOJU');
942             end;
943             '2' : begin
944                 writeln('          ZUZYCIE MPS NA DRODZE  OD REJONU POSTOJU DO REJONU DZIENNEGO ODPOCZYNKU
945             ');
946             end;
947             '3' : begin
948                 writeln('          ZUZYCIE MPS W CZASIE CALEGO MARSZU');
949             end;
950         writeln;
951         LowVideo;
952         write(' Kolumna ');
953         NormVideo;
954         write(' ');
955         LowVideo;
956         write(' ZuBS    ZuONPK    ZuONPG    ZuONCZ    sumaON');
957         gotoxy(62,5);
958         write(' KBS = ',KBS:4:2);
959         gotoxy(62,6);
960         write(' KONPK = ',KONPK:4:2);
961         gotoxy(62,7);
962         write(' KONPG = ',KONPG:4:2);
963         gotoxy(62,8);
964         write(' KONCZ = ',KONCZ:4:2);
965         gotoxy(62,10);
966         write(' L1    = ',L1:5:1,' km');
967         gotoxy(62,11);
968         write(' L2    = ',L12-L1:5:1,' km');
969         gotoxy(62,12);
970         write(' L12   = ',L12:5:1,' km');
971     end;
972     gotoxy(5,nr_linii);
973     write(k:2);
974     gotoxy(12,nr_linii);
975     if ZuBS > 0 then
976         write(ZuBS:6:2) else write('-----');
977     gotoxy(22,nr_linii);
978     if ZuONPK > 0 then
979         write(ZuONPK:6:2) else write('-----');
980     gotoxy(32,nr_linii);
981     if ZuONPG > 0 then
982         write(ZuONPG:6:2) else write('-----');
983     gotoxy(42,nr_linii);
984     if ZuONCZ > 0 then
985         write(ZuONCZ:6:2) else write('-----');
986     suma:=ZuONPK+ZuONPG+ZuONCZ;
987     gotoxy(51,nr_linii);
988     if suma > 0 then
989         write(suma:6:2) else write('-----');
```

```

990         end;
991     end;
992     writeln;
993     writeln('-----');
994     write(' suma');
995     gotoxy(11,nr_linii+2);
996     if (sBS > 0) then write(sBS:7:2) else write('-----');
997     gotoxy(21,nr_linii+2);
998     if (sONPK > 0) then write(sONPK:7:2) else write('-----');
999     gotoxy(31,nr_linii+2);
1,000     if (sONPG > 0) then write(sONPG:7:2) else write('-----');
1,001     gotoxy(41,nr_linii+2);
1,002     if (sONCZ > 0) then write(sONCZ:7:2) else write('-----');
1,003     gotoxy(50,nr_linii+2);
1,004     sON:=sONPK+sONPG+sONCZ;
1,005     if (sON > 0) then write(sON:7:2) else write('-----');
1,006     czekanie_na_return;
1,007     end;
1,008     until ((polecenie = '4') or (polecenie = '5'));
1,009     until polecenie = '5';
1,010     end
1,011     else
1,012         begin
1,013             gotoxy(19,18);
1,014             writeln('Nie sformowano kolumn',chr(7));
1,015             czekanie_na_return;
1,016             nie_wym:=true;
1,017         end;
1,018     end;
1,019
1,020 Procedure obliczenie_masy_jn;
1,021     var
1,022         byla_kolumna : Boolean;
1,023         nr_od,nr_linii,akt_oddzial: integer;
1,024         pr : real;
1,025     begin
1,026         nr_linii:=3;
1,027         byla_kolumna:=False;
1,028         for k:=1 to ilosc_kol do
1,029             begin
1,030                 MBS:=0;
1,031                 MONPK:=0;
1,032                 MONPG:=0;
1,033                 MONCZ:=0;
1,034                 if kolumny[k,1].nr_odd > 0 then
1,035                     begin
1,036                         if nr_linii = 3 then ClrScr;
1,037                         nr_linii:=nr_linii+2;
1,038                         byla_kolumna:=true;
1,039                         akt_oddzial:=1;
1,040                         while kolumny[k,akt_oddzial].nr_odd > 0 do
1,041                             begin
1,042                                 nr_od:=kolumny[k,akt_oddzial].nr_odd;
1,043                                 ul:=kolumny[k,akt_oddzial].ulamek;
1,044                                 pr:=kolumny[k,akt_oddzial].proc/100.0;

```

```
1,045         with tab_jed_nap[nr_od] do
1,046             begin
1,047                 MBS:= MBS+ BS*ul*pr;
1,048                 MONPK:=MONPK+ONPK*ul*pr;
1,049                 MONPG:=MONPG+ONPG*ul*pr;
1,050                 MONCZ:=MONCZ+ONCZ*ul*pr;
1,051             end;
1,052             akt_oddzial:=akt_oddzial+1;
1,053         end;
1,054         gotoxy(3,nr_linii);
1,055         write(' ',k);
1,056         gotoxy(18,nr_linii);
1,057         if MBS > 0 then write(MBS:6:2)
1,058             else write('-----');
1,059         gotoxy(26,nr_linii);
1,060         if MONPK > 0 then write(MONPK:6:2)
1,061             else write('-----');
1,062         gotoxy(36,nr_linii);
1,063         if MONPG > 0 then write(MONPG:6:2)
1,064             else write('-----');
1,065         gotoxy(47,nr_linii);
1,066         if MONCZ > 0 then write(MONCZ:6:2)
1,067             else write('-----');
1,068         gotoxy(54,nr_linii);
1,069         suma:=MONPK+MONPG+MONCZ;
1,070         if suma > 0 then write(suma:6:2)
1,071             else write('-----');
1,072     end;
1,073 end;
1,074 if not byla_kolumna then
1,075     begin
1,076         nie_wym:=true;
1,077         gotoxy(20,20);
1,078         write(' Nie sformowano kolumn ',chr(7));
1,079     end
1,080 else
1,081     begin
1,082         gotoxy(8,1);
1,083         writeln(' ZESTAWIENIE MASY JEDNOSTEK NAPELNIENIA KOLUMN ');
1,084         LowVideo;
1,085         writeln;
1,086         write(' Kolumny ');
1,087         gotoxy(16,3);
1,088         writeln('   BS       ONPK       ONPG       ONCZ   sumaON   ');
1,089         NormVideo;
1,090         czekanie_na_return;
1,091     end;
1,092 end;
1,093
1,094 Procedure zapis_jedn_nap;
1,095     var
1,096         FilVar : file of jedn_nap;
1,097
1,098     begin
1,099         Assign(FilVar,'jedn_nap');
```

```
1,100      {$I-}
1,101      rewrite(FilVar);
1,102      {$I+}
1,103      if (IoResult=0) then
1,104          begin
1,105              for k:=1 to ilosc_podod do
1,106                  Write(FilVar,tab_jed_nap[k]);
1,107                  Close(FilVar);
1,108          end
1,109      else
1,110          begin
1,111              gotoxy(22,15);
1,112              Writeln('Brak zbioru JEDN_NAP',chr(7));
1,113              czekanie_na_return;
1,114          end;
1,115      end;
1,116      Procedure zapis_cystern;
1,117      var
1,118          FilVar : file of opis_cyst;
1,119      begin
1,120          Assign(FilVar,'cysterny');
1,121          {$I-}
1,122          Rewrite(FilVar);
1,123          {$I+}
1,124          if (ioresult=0) then
1,125              begin
1,126                  for k:=1 to ilosc_sr do
1,127                      Write(FilVar,cysterny[k]);
1,128                      Close(FilVar);
1,129              end;
1,130          end;
1,131      Procedure odczyt_cystern;
1,132      var
1,133          FilVar : file of opis_cyst;
1,134      begin
1,135          Assign(FilVar,'cysterny');
1,136          {$I-}
1,137          Reset(FilVar);
1,138          {$I+}
1,139          if (ioresult=0) then
1,140              begin
1,141                  for k:=1 to ilosc_sr do
1,142                      Read(FilVar,cysterny[k]);
1,143                      Close(FilVar);
1,144                      sa_cysterny:=true;
1,145                      ilosc_benzyny(ilosc_sr);
1,146              end
1,147          else
1,148              begin
1,149                  gotoxy(22,15);
1,150                  writeln('Brak zbioru CYSTERNY',chr(7));
1,151                  czekanie_na_return;
1,152              end;
1,153          end;
1,154      Procedure odczyt_jedn_nap;
```

```
1,155 var
1,156   FilVar : file of jedn_nap;
1,157 begin
1,158   Assign(FilVar,'jedn_nap');
1,159   {$I-}
1,160   Reset(FilVar);
1,161   {$I+}
1,162   if (ioresult=0) then
1,163     begin
1,164       for k:=1 to ilosc_podod do
1,165         Read(FilVar,tab_jed_nap[k]);
1,166       Close(FilVar);
1,167     end;
1,168 end;
1,169
1,170 Procedure odczyt_skl_kol;
1,171 var
1,172   FilVar : file of skl_kol;
1,173 begin
1,174   Assign(FilVar,'kolumnym');
1,175   {$I-}
1,176   Reset(FilVar);
1,177   {$I+}
1,178   if (ioresult=0) then
1,179     begin
1,180       for k:=1 to ilosc_kol do
1,181         for j:=1 to ilosc_podod do
1,182           Read(FilVar,kolumny[k,j]);
1,183       Close(FilVar);
1,184     end
1,185   else
1,186     begin
1,187       gotoxy(22,15);
1,188       Writeln('Brak zbioru KOLUMNYM',chr(7));
1,189       czekanie_na_return;
1,190     end;
1,191 end;
1,192
1,193
1,194 Procedure zapis_skl_kol;
1,195 var
1,196   FilVar : file of skl_kol;
1,197 begin
1,198   Assign(FilVar,'kolumnym');
1,199   {$I-}
1,200   Rewrite(FilVar);
1,201   {$I+}
1,202   if (ioresult=0) then
1,203     begin
1,204       for k:=1 to ilosc_kol do
1,205         for j:=1 to ilosc_podod do
1,206           Write(FilVar,kolumny[k,j]);
1,207       Close(FilVar);
1,208     end;
1,209 end;
```

```
1,210 Procedure operacje_dyskowe;
1,211   begin
1,212     ClrScr;
1,213     gotoxy(24,2);
1,214     LowVideo;
1,215     Writeln(' OPERACJE DYSKOWE ');
1,216     NormVideo;
1,217     writeln;
1,218     Writeln('       1. Zapis tabeli jednostek napelnienia');
1,219     writeln;
1,220     Writeln('       2. Odczyt tabeli jednostek napelnienia');
1,221     writeln;
1,222     Writeln('       3. Zapis skladu kolumn');
1,223     writeln;
1,224     Writeln('       4. Odczyt skladu kolumn');
1,225     writeln;
1,226     Writeln('       5. Zapis srodkow sluzby mps');
1,227     writeln;
1,228     Writeln('       6. Odczyt srodkow sluzby mps');
1,229     writeln;
1,230     Writeln('       7. Powrot do glownego menu');
1,231     writeln;
1,232     write ('           polecenie - ');
1,233     repeat
1,234       for k:=19 to 24 do
1,235         begin
1,236           gotoxy(1,k);
1,237           ClrEol;
1,238         end;
1,239     gotoxy(38,18);
1,240     write(' ');
1,241     gotoxy(38,18);
1,242     repeat
1,243       read(polecenie);
1,244     until ((polecenie>'0') and (polecenie<'8'));
1,245     case polecenie of
1,246       '1' : zapis_jedn_nap;
1,247       '2' : odczyt_jedn_nap;
1,248       '3' : zapis_skl_kol;
1,249       '4' : odczyt_skl_kol;
1,250       '5' : zapis_cystern;
1,251       '6' : odczyt_cystern;
1,252     end;
1,253     until polecenie='7';
1,254   end;
1,255
1,256 Procedure wysw_oddzial(nr_kol,akt_oddzial : Integer);
1,257 var
1,258   nr_linii, pos_w_linii : Integer;
1,259
1,260   begin
1,261     if kolumny[nr_kol,akt_oddzial].nr_odd <> 0 then
1,262       begin
1,263         nr_linii:=(akt_oddzial-1) div 3;
1,264         pos_w_linii:=(akt_oddzial-1) mod 3;
```

```
1,265     GoToxy(pos_w_linii*27+1,nr_linii+15);
1,266     if kolumny[nr_kol,akt_oddzial].ulamek <> 1 then
1,267         Write(kolumny[nr_kol,akt_oddzial].wlk_odd);
1,268         Write(nazwa_podod[kolumny[nr_kol,akt_oddzial].nr_odd]);
1,269     end;
1,270 end;
1,271
1,272 Function cyfra( znak : char ) : boolean;
1,273 begin
1,274     cyfra:=False;
1,275     if ((znak >= '0') and (znak <= '9')) then cyfra:=True;
1,276 end;
1,277
1,278 Procedure wysw_nazwy;
1,279 var
1,280     i,j : Integer;
1,281 begin
1,282     for k:=1 to 10 do
1,283         begin
1,284             GoToxy(1,k);
1,285             ClrEol;
1,286         end;
1,287     Clrscr;
1,288     GoToxy(2,1);
1,289     LowVideo;
1,290     WriteLn('          DOPUSZCZALNE    NAZWY    PODODDZIALOW    ');
1,291     ClrEol;
1,292     NormVideo;
1,293     WriteLn;
1,294     ClrEol;
1,295     for k:=1 to ilosc_podod do
1,296         begin
1,297             i:=(k-1) div 7;
1,298             j:=(k-1) mod 7;
1,299             GoToxy(j*12+1,i+3);
1,300             Write(nazwa_podod[k]);
1,301         end;
1,302     end;
1,303
1,304 function oblicz_numer(naz : nazwa) : integer;
1,305 begin
1,306     if naz[0]=chr(0) then
1,307         begin
1,308             oblicz_numer:=100;
1,309             exit;
1,310         end;
1,311     for k:=1 to ilosc_podod do
1,312         begin
1,313             if ((pos(naz,nazwa_podod[k]) <> 0) and (pos(nazwa_podod[k],naz) <> 0))then
1,314                 begin
1,315                     oblicz_numer:=k;
1,316                     exit;
1,317                 end;
1,318         end;
1,319     oblicz_numer:=0;
```

```
1,320 end;
1,321
1,322 Procedure utw_kolumne;
1,323 var
1,324     numer_odd,nr_kol,akt_oddzial : integer;
1,325     procent : real;
1,326
1,327 begin
1,328     ClrScr;
1,329     repeat
1,330         gotoxy(1,12);
1,331         write(' Podaj numer kolumny : ');
1,332         nr_kol:=pobranie_numeru(25,12);
1,333     until ((nr_kol > 0) and (nr_kol <= ilosc_kol));
1,334     writeln;
1,335     wysw_nazwy;
1,336     gotoxy(1,8);
1,337     LowVideo;
1,338     writeln(' Kolumna nr : ',nr_kol);
1,339     NormVideo;
1,340     writeln;
1,341     writeln(' Wprowadz kolejna nazwe pododdzialu: ');
1,342     akt_oddzial := 1;
1,343     repeat
1,344         gotoxy(40,12);
1,345         ClrEol;
1,346         gotoxy(40,10);
1,347         ClrEol;
1,348         readln(nazwa_odd);
1,349         if cyfra(nazwa_odd[1]) and cyfra(nazwa_odd[3]) and (nazwa_odd[2] = '/') then
1,350             begin
1,351                 kolumny[nr_kol,akt_oddzial].wik_odd:=copy(nazwa_odd,1,3);
1,352                 cyfra_1:=ord(nazwa_odd[1])-48;
1,353                 cyfra_2:=ord(nazwa_odd[3])-48;
1,354                 kolumny[nr_kol,akt_oddzial].ulamek:=cyfra_1/cyfra_2;
1,355                 nazwa_odd:=copy(nazwa_odd,4,10);
1,356             end
1,357         else
1,358             kolumny[nr_kol,akt_oddzial].ulamek:=1;
1,359             numer_odd:=oblicz_numer(nazwa_odd);
1,360             if ((numer_odd > 0) and (numer_odd <= ilosc_podod)) then
1,361                 begin
1,362                     kolumny[nr_kol,akt_oddzial].nr_odd:=numer_odd;
1,363                     repeat
1,364                         writeln;
1,365                         writeln(' Podaj procent ukompletowania : ');
1,366                         procent:=100;
1,367                         gotoxy(40,12);
1,368                         ClrEol;
1,369                         readln(procent);
1,370                     until ((procent >= 0) and (procent <= 100));
1,371                     kolumny[nr_kol,akt_oddzial].proc:=procent;
1,372                     wysw_oddzial(nr_kol,akt_oddzial);
1,373                     akt_oddzial:=akt_oddzial+1;
1,374                 end;
end;
```

```
1,375   until ((numer_odd = 100) or (akt_oddzial > ilosc_podod));
1,376   for k:=akt_oddzial to ilosc_podod do
1,377       kolumny[nr_kol,k].nr_odd:=0;
1,378   end;
1,379
1,380 Procedure czytaj_liczbe(xpos,ypos : Integer; var Zmienna : Real);
1,381   var
1,382       code   : Integer;
1,383       liczba  : string[6];
1,384   begin
1,385       repeat
1,386           GoToxy(xpos,ypos);
1,387           ClrEol;
1,388           ReadLn(liczba);
1,389           Val(liczba,Zmienna,code);
1,390       until code=0;
1,391   end;
1,392
1,393 Procedure zestawienie_masy;
1,394   var
1,395       n : integer;
1,396   begin
1,397       repeat
1,398           ClrScr;
1,399           GoToxy(1,1);
1,400           WriteLn(' ZESTAWIENIE MASY JEDNOSTEK NAPELNIENIA PODODDZIALOW FULKU');
1,401           LowVideo;
1,402           writeln;
1,403           Write(' Oddzial ');
1,404           NormVideo;
1,405           Write(' ');
1,406           Write(' BS      ONPK      ONPG      ONCZ      suma ON ');
1,407           NormVideo;
1,408           writeln;
1,409           n:=3;
1,410           for k:=1 to ilosc_podod do
1,411               begin
1,412                   gotoxy(2,n+2);
1,413                   Write(nazwa_podod[k]);
1,414                   GoToxy(16,n+2);
1,415                   if tab_jed_nap[k].BS > 0 then
1,416                       Write(tab_jed_nap[k].BS:4:2)
1,417                   else
1,418                       Write('----');
1,419                   GoToxy(23,n+2);
1,420                   if tab_jed_nap[k].ONPK > 0 then
1,421                       Write(tab_jed_nap[k].ONPK:4:2)
1,422                   else
1,423                       Write('----');
1,424                   GoToxy(32,n+2);
1,425                   if tab_jed_nap[k].ONPG > 0 then
1,426                       Write(tab_jed_nap[k].ONPG:4:2)
1,427                   else
1,428                       Write('----');
1,429                   GoToxy(41,n+2);
```

```
1,430     if tab_jed_nap[k].ONCZ > 0 then
1,431         Write(tab_jed_nap[k].ONCZ:4:2)
1,432     else
1,433         Write('----');
1,434         GoToxy(50,n+2);
1,435         suma:=tab_jed_nap[k].ONCZ+tab_jed_nap[k].ONPK+tab_jed_nap[k].ONPG;
1,436         if suma > 0 then
1,437             Write(suma:5:2)
1,438         else
1,439             Write('----');
1,440         Writeln;
1,441         n:=n+1;
1,442     end;
1,443 writeln;
1,444 Write('    Czy chcesz zmienic jn pododdzialu (t/n) ? ');
1,445 repeat
1,446     Read(polecenie);
1,447     polecenie:=UpCase(polecenie);
1,448 until ((polecenie='T') or (polecenie='N'));
1,449     if polecenie='T' then
1,450         begin
1,451             GoToxy(60,10);
1,452             Write('Podaj nazwe oddzialu');
1,453             repeat
1,454                 GoToxy(60,11);
1,455                 ClrEol;
1,456                 ReadLn(Nazwa_odd);
1,457                 numer_odd:=oblicz_numer(Nazwa_odd);
1,458             until ((numer_odd > 0) and (numer_odd < ilosc_podod));
1,459             GoToxy(60,12);
1,460             Write('Podaj nowe jn :');
1,461             GoToxy(60,13);
1,462             Write('BS    ');
1,463             czytaj_liczbe(67,13,tab_jed_nap[numer_odd].BS);
1,464             GoToxy(60,14);
1,465             Write('ONPK  ');
1,466             czytaj_liczbe(67,14,tab_jed_nap[numer_odd].ONPK);
1,467             GoToxy(60,15);
1,468             Write('ONPG  ');
1,469             czytaj_liczbe(67,15,tab_jed_nap[numer_odd].ONPG);
1,470             GoToxy(60,16);
1,471             Write('ONCZ  ');
1,472             czytaj_liczbe(67,16,tab_jed_nap[numer_odd].ONCZ);
1,473         end;
1,474     until polecenie='N';
1,475 end;
1,476
1,477 Procedure przegladanie_kolumn;
1,478 var
1,479     nr_kol,akt_oddzial : Integer;
1,480     byla_kol           : Boolean;
1,481
1,482 begin
1,483     byla_kol:=False;
1,484     ClrScr;
```

```
1,485 GoToxy(25,10);
1,486 Write('Podaj numer kolumny ');
1,487 repeat
1,488   nr_kol:=pobranie_numeru(47,10);
1,489 until ((nr_kol > 0) and (nr_kol <=ilosc_kol));
1,490 ClrScr;
1,491 if kolumny[nr_kol,1].nr_odd<>0 then
1,492   for akt_oddzial:=1 to ilosc_podod do
1,493     if kolumny[nr_kol,akt_oddzial].nr_odd<>0 then
1,494       begin
1,495         wysw_oddzial(nr_kol,akt_oddzial);
1,496         byla_kol:=True;
1,497       end;
1,498 if byla_kol then
1,499   begin
1,500     GoToxy(25,12);
1,501     LowVideo;
1,502     WriteLn(' SKLAD KOLUMNY NR ',nr_kol,' ');
1,503     NormVideo;
1,504   end
1,505 else
1,506   begin
1,507     GoToxy(18,18);
1,508     Write(' Nie sformowano kolumny ',nr_kol,chr(7));
1,509   end;
1,510   czekanie_na_return;
1,511 end;
1,512
1,513 Procedure menu_form;
1,514 begin
1,515   for k:=1 to 10 do
1,516     begin
1,517       GoToxy(1,k);
1,518       ClrEol;
1,519     end;
1,520   LowVideo;
1,521   GoToxy(28,3);
1,522   WriteLn(' FORMOWANIE KOLUMN ');
1,523   Normvideo;
1,524   Gotoxy(20,5);
1,525   Write(' 1. Przeglądanie istniejących kolumn');
1,526   GoToxy(20,6);
1,527   Write(' 2. Sformowanie nowej kolumny');
1,528   GoToxy(20,7);
1,529   Write(' 3. Powrot do glownwgo menu');
1,530   GoToxy(32,9);
1,531   Write('Polecenie - ');
1,532 end;
1,533
1,534 Procedure sformowanie_kolumn;
1,535 begin
1,536   ClrScr;
1,537   menu_form;
1,538   repeat
1,539     for k:=10 to 24 do
```

```
1,540     begin
1,541         GoToxy(1,k);
1,542         ClrEol;
1,543     end;
1,544     gotoxy(44,9);
1,545     write(' ');
1,546     GoToxy(44,9);
1,547     repeat
1,548         Read(polecenie);
1,549         polecenie:=UpCase(polecenie);
1,550     until ((polecenie > '0') and (polecenie < '4'));
1,551     case polecenie of
1,552     '1' : begin
1,553         przegladanie_kolumn;
1,554         menu_form;
1,555     end;
1,556     '2' : begin
1,557         utw_kolumne;
1,558         menu_form;
1,559     end;
1,560     end;
1,561     until (polecenie = '3');
1,562 end;
1,563
1,564 Procedure koniec_pracy;
1,565 begin
1,566     ClrScr;
1,567     Halt;
1,568 end;
1,569
1,570 Procedure wyswietlenie_menu;
1,571 begin
1,572     ClrScr;
1,573     GoToxy(8,3);
1,574     WriteLn(' PROGRAM ANALIZY ZUZYCIA I SPOSOBU PRZYDZIAU MPS ');
1,575     GoToxy(2,5);
1,576     WriteLn(' Głowne menu : ');
1,577     gotoxy(2,6);
1,578     writeln(' ----- ');
1,579     NormVideo;
1,580     WriteLn;
1,581     WriteLn(' 1. Zestawienie masy jednostek napelnienia pododdzialow');
1,582     writeln;
1,583     WriteLn(' 2. Sformowanie kolumny marszowej');
1,584     writeln;
1,585     WriteLn(' 3. Obliczenie jednostek napelnienia poszczegolnych kolumn');
1,586     writeln;
1,587     WriteLn(' 4. Obliczenie zuzycia mps przez poszczegolne kolumny');
1,588     writeln;
1,589     WriteLn(' 5. Podzial posiadanych sil i srodkow sluzby mps');
1,590     writeln;
1,591     WriteLn(' 6. Operacje dyskowe');
1,592     writeln;
1,593     WriteLn(' 7. Zakonczenie pracy z programem');
1,594     WriteLn;
```

```

1,595   writeLn;
1,596   writeLn;
1,597   Write('           Polecenie - ');
1,598   end;
1,599
1,600 Procedure inicjalizacja(odd_nr : Integer; oddzial : Nazwa; jnBS,jnONPK,jnONPG,jnONCZ : Real);
1,601 begin
1,602   nazwa_podod[odd_nr]:=oddzial;
1,603   with tab_jed_nap[odd_nr] do
1,604     begin
1,605       BS:=jnBS;
1,606       ONPK:=jnONPK;
1,607       ONPG:=jnONPG;
1,608       ONCZ:=jnONCZ;
1,609     end;
1,610 end;
1,611 begin
1,612   sa_cysterny:=False;
1,613   cysterny[1].naz:='samochod transportowy a beczkami STRB';
1,614   cysterny[1].pojem:=3.6;
1,615   cysterny[2].naz:='cysterna dystrybutor 4.5      CD4.5';
1,616   cysterny[2].pojem:=4.5;
1,617   cysterny[3].naz:='cysterna dystrybutor 5      CD5W';
1,618   cysterny[3].pojem:=5;
1,619   cysterny[4].naz:='cyst. dyst. paliwowo-olejowy  CDPO4W';
1,620   cysterny[4].pojem:=4;
1,621   cysterny[5].naz:='cysterna dystrybutor 7.5    CD7.5';
1,622   cysterny[5].pojem:=7.5;
1,623   cysterny[6].naz:='samochod transportowy z kanistr. STRK';
1,624   cysterny[6].pojem:=3.6;
1,625   cysterny[7].naz:='cysterna dystrybutor      CD5WK';
1,626   cysterny[7].pojem:=4.5;
1,627   cysterny[8].naz:='cysterna przyczepa          CP';
1,628   cysterny[8].pojem:=3;
1,629   cysterny[9].naz:='przyczepa transportowa z beczkami PT';
1,630   cysterny[9].pojem:=3;
1,631   cysterny[10].naz:='cysterna z przyczepa      CD7.5+CP11';
1,632   cysterny[10].pojem:=16.5;
1,633   for k:=1 to ilosc_kol do
1,634     kolumny[k,1].nr_odd:=0;
1,635   ClrScr;
1,636   HighVideo;
1,637   GoToxy(5,3);
1,638   Write(chr($c9));
1,639   for k:=1 to 70 do Write(chr($cd));
1,640   WriteLn(chr($bb));
1,641   WriteLn('           ',chr($ba));
1,642   WriteLn('           ',chr($ba));
1,643   WriteLn('           ',chr($ba));
1,644   WriteLn('           ',chr($ba));
1,645   WriteLn('           ',chr($ba));
1,646   WriteLn('           ',chr($ba));
1,647   WriteLn('           ',chr($ba));
1,648   WriteLn('           ',chr($ba));
1,649   WriteLn('           ',chr($ba));

```

CENTRUM INFORMATYKI AON
KATEDRY LOGISTYKI WYDZIAŁU WNGJSK LADOWYCH AON

```

prezentuje program :

```

```

1,650 WriteLn(' ',chr($ba),' ',chr($ba));
1,651 WriteLn(' ',chr($ba),' ',chr($ba));
1,652 WriteLn(' ',chr($ba),' ZAOPATRYWANIE W MPS pz WYKONUJACEGO MARSZ ',chr($ba));
1,653 WriteLn(' ',chr($ba),' ',chr($ba));
1,654 WriteLn(' ',chr($ba),' ',chr($ba));
1,655 WriteLn(' ',chr($ba),' ',chr($ba));
1,656 WriteLn(' ',chr($ba),' ',chr($ba));
1,657 WriteLn(' ',chr($ba),' opracowany przez : ',chr($ba));
1,658 WriteLn(' ',chr($ba),' ',chr($ba));
1,659 WriteLn(' ',chr($ba),' pplk dr inz. Zbigniew KLIMKIEWICZ ',chr($ba));
1,660 WriteLn(' ',chr($ba),' plk dr Aleksander LOJKO ',chr($ba));
1,661 WriteLn(' ',chr($ba),' ',chr($ba));
1,662 WriteLn(' ',chr($cB));
1,663 for k:=1 to 70 do Write(chr($cd));
1,664 WriteLn(chr($bc));
1,665 WriteLn;
1,666 NormVideo;
1,667 WriteLn(' ',chr($cB),' naciśnij klawisz ENTER');
1,668 Read(polecenie);
1,669 inicjalizacja( 1,'bcz/T-72/', 1.27, 0.80, 0.50,33.52);
1,670 inicjalizacja( 2,'bcz/T-55/', 1.27, 0.80, 0.50,26.80);
1,671 inicjalizacja( 3,'bpz', 2.46, 1.52, 0.46, 0);
1,672 inicjalizacja( 4,'bpzmot', 2.46, 1.52, 0.46, 0);
1,673 inicjalizacja( 5,'da', 2.60, 1.08, 0.47, 0);
1,674 inicjalizacja( 6,'das', 1.85, 1.36, 6.67, 0);
1,675 inicjalizacja( 7,'daplot', 2.05, 2.86, 2.05, 0);
1,676 inicjalizacja( 8,'bappanc', 1.81, 0, 0, 0);
1,677 inicjalizacja( 9,'kr', 1.30, 0, 0, 0);
1,678 inicjalizacja(10,'ksap', 1.04, 1.39, 4.22, 0);
1,679 inicjalizacja(11,'kl', 0.50, 1.50, 0, 0);
1,680 inicjalizacja(12,'kmed', 1.04, 0, 0, 0);
1,681 inicjalizacja(13,'kzaop', 6.28, 6.56, 0, 0);
1,682 inicjalizacja(14,'krem', 8.35, 1.66, 2.66, 0);
1,683 inicjalizacja(15,'plochem', 0.72, 0.39, 0, 0);
1,684 inicjalizacja(16,'ploirr', 0.49, 0.26, 0, 0);
1,685 inicjalizacja(17,'SD', 0.25, 0.15, 0, 0);
1,686 inicjalizacja(18,'CKWL', 0.30, 0.00, 0.00, 0.00);
1,687 nie_wym:=False;
1,688 repeat
1,689 if not nie_wym then wyswietlenie_menu
1,690 else
1,691 begin
1,692 for k:=16 to 24 do
1,693 begin
1,694 GoToxy(1,k);
1,695 ClrEol;
1,696 end;
1,697 end;
1,698 GoToxy(38,24);
1,699 repeat
1,700 Read(polecenie);
1,701 until ((polecenie > '0') and (polecenie < '8'));
1,702 nie_wym:=False;
1,703 case polecenie of
1,704 '1' : zestawienie_masy;

```

```
1,705      '2' : sformowanie_kolumn;  
1,706      '3' : obliczenie_masy_jn;  
1,707      '4' : obliczenie_zuzycia;  
1,708      '5' : podzial_srodkow;  
1,709      '6' : operacje_dyskowe;  
1,710      '7' : koniec_pracy;  
1,711      end;  
1,712      until False;  
1,713      end.  
1,714
```

