

Języki formalne i metody kompilacji

Laboratorium nr 10

Stefan Sokołowski

Zadanie 1:

Napisać w języku wewnętrznym program wpisujący do tablicy kolejne liczby naturalne; czyli przetłumaczyć na język wewnętrzny poniższą komendę z C:

```
for (i=0; i<n; i++) a[i]=i;
```

Można założyć, że

- na zmienną `i` przeznaczona jest komórka o adresie 200,
- na zmienną `n` przeznaczona jest komórka o adresie 201 i chwilowo wpisana jest tam liczba 100,
- elementy tablicy `a` zapisane są w kolejnych komórkach począwszy od adresu 202.

Wskazówka:

Użyć modyfikacji adresowych (patrz: wykład 9¹).

Zadanie 2:

Napisać w języku wewnętrznym program obliczający sumę danego ciągu liczb.

Zadanie 3:

Napisać w języku wewnętrznym program obliczający iloczyn skalarny dwóch danych wektorów równej długości.

Zadanie 4:

Napisać w języku wewnętrznym program obliczający iloczyn skalarny dwóch przekątnych tablicy dwuwymiarowej kwadratowej, czyli wielkość

$$a[0][0]*a[0][n-1] + a[1][1]*a[1][n-2] + \dots + a[n-1][n-1]*a[n-1][0]$$

¹iis.ans-elblag.pl/~stefan/Dydaktyka/JezForm/Wyklady/09a.pdf

Zadanie 5:

Proszę napisać i uruchomić program w języku „wewnętrznym”, dodający dwa wektory: $\vec{w} = \vec{x} + \vec{y}$. Program ma działać poprawnie dla wektorów dowolnej (ale oczywiście tej samej) długości umieszczonych pod dowolnym adresem gdziekolwiek w pamięci.

```
1:      ... # n
2:      ... # adr_x
3:      ... # adr_y
4:      ... # adr_w
... komendy programu ...
```

Program powinien być skonstruowany według sugestii z powyższego schematu; t.zn. powinien mieć miejsce na wpisanie wspólnej długości n wektorów, adresu początku wektora x , adresu początku wektora y oraz adresu, od którego ma być wpisany wektor wynikowy w . I musi być tak skonstruowany, żeby jego zastosowanie do wektorów innej długości zapisanych w innych miejscach pamięci wymagało wyłącznie zmiany wartości komórek 1–4, a nie wymagało zmiany niczego w samym programie.

Wskazówka:

Jeśli w komórce o numerze nr wpisana jest liczba adr , to komenda `loada 0+[nr]` powoduje załadowanie do rejestru arytmetycznego zawartości komórki o numerze adr . To umożliwia realizację „wskaźników” z języka C, których odpowiedniki potrzebne są w tym zadaniu.

Zadanie 6 (domowe specjalne):

Napisać w języku wewnętrznym program obliczający iloczyn dwóch macierzy dowolnych (danych przez użytkownika) rozmiarów.

Uwaga:

Pierwszej osobie, która przyśle mi (na s.sokolowski@ans-elblag.pl) poprawne rozwiązanie tego zadania, przysługują punkty do zaliczenia laboratorium. Liczba tych punktów zależy od oryginalności i elegancji rozwiązania.

Zadanie 7:

Program z zad. 5 łatwo jest przerobić na taki, który w ciągu wynikowym (jego adres ma być podawany w komórce `adr_w`) umieszcza nie sumy tylko iloczynów współrzędnych wektorów \vec{x} i \vec{y} :

$$x_0 \cdot y_0, \quad x_1 \cdot y_1, \quad \dots, \quad x_{n-1} \cdot y_{n-1}$$

Proszę

- dokonać takiej właśnie przeróbki programu z zad. 5;
- zrobić z niego podprogram (funkcję), który może wywołać dowolny program główny i który po zakończeniu obliczenia powraca do programu głównego;
- napisać program liczący iloczyn skalarny dwóch wektorów

$$x_0 \cdot y_0 + x_1 \cdot y_1 + \dots + x_{n-1} \cdot y_{n-1}$$

w taki sposób, że **najpierw** wywołuje powyższy podprogram dla policzenia iloczynów współrzędnych, a **potem** oblicza sumę ciągu iloczynów (por. zad. 4 z lab. 9).

Zadanie 8 (czy liczba jest pierwsza?):

Proszę napisać w języku „wewnętrznym” program, który sprawdza, czy dana liczba n jest pierwsza. Dokładniej:

- program powinien przeglądać liczby od 2 do $n - 1$ w poszukiwaniu dzielnika liczby n ; jeśli znajdzie, to zakończyć działanie z odpowiedzią, że liczba jest złożona, a jeśli dojdzie do końca nie znalazłszy dzielnika, to odpowiadać, że jest pierwsza;
- do sprawdzenia podzielności program powinien skorzystać z podprogramu omówionego na wykładzie²; dla sprawdzenia podzielności n przez k przy pomocy tego podprogramu należy:
 1. do komórki 1 włożyć n ,
 2. do komórki 2 włożyć k ,
 3. do komórki 0 włożyć adres komendy do wykonania po wykonaniu dzielenia (pkt. 5 poniżej),
 4. skoczyć do komórki 5,
 5. sprawdzić, czy zawartość komórki 4 (reszta z dzielenia) jest zerowa;
- jeśli liczba jest pierwsza, to niech program do komórki oznaczającej **wynik** włoży jedynkę, a jeśli jest złożona, to zero.

Zadanie 9 (domowe na punkty):

Napisać w języku „wewnętrzny” program, który dowolną dodatnią liczbę całkowitą zamienia na liczbę ósemkową; np.

$$\begin{array}{ll}
 10 \rightarrow \mathbf{12} & \text{bo } 10 = \mathbf{1} \cdot 8 + \mathbf{2} \\
 19 \rightarrow \mathbf{23} & \text{bo } 19 = \mathbf{2} \cdot 8 + \mathbf{3} \\
 355 \rightarrow \mathbf{543} & \text{bo } 355 = (\mathbf{5} \cdot 8 + \mathbf{4}) \cdot 8 + \mathbf{3}
 \end{array}$$

Do wyliczania kolejnych cyfr ósemkowych potrzebne jest dzielenie z resztą. Proszę do tego użyć tego samego podprogramu na dzielenie.

Uwaga:

Czas na przysyłanie (na s.sokolowski@ans-elblag.pl): 2 tygodnie od zajęć z lab.10 — potem już nie przyjmuję rozwiązań. Liczba punktów zależy od oryginalności i elegancji rozwiązania.

Zadanie 10 (domowe specjalne):

Gotowy program, stanowiący rozwiązanie zadania 7, proszę przerobić na podprogram przez zastąpienie komendy **stop** na skok do zapamiętanego wcześniej adresu powrotu. Następnie użyć tego podprogramu do zrealizowania obliczania iloczynu macierzy wymiaru $n \times k$ przez macierz wymiaru $k \times m$ dla dowolnych liczb dodatnich n, k, m . Zmiana tych rozmiarów albo miejsca macierzy w pamięci nie może wymagać żadnych zmian w samym kodzie programu.

Wskazówka:

Przypominam, że w iloczynie macierzy

$$\begin{bmatrix} A_{0,0} & A_{0,1} & \dots & A_{0,k-1} \\ A_{1,0} & A_{1,1} & \dots & A_{1,k-1} \\ \dots & & & \\ A_{n-1,0} & A_{n-1,1} & \dots & A_{n-1,k-1} \end{bmatrix} \cdot \begin{bmatrix} B_{0,0} & B_{0,1} & \dots & B_{0,m-1} \\ B_{1,0} & B_{1,1} & \dots & B_{1,m-1} \\ \dots & & & \\ B_{k-1,0} & B_{k-1,1} & \dots & B_{k-1,m-1} \end{bmatrix} = \begin{bmatrix} C_{0,0} & C_{0,1} & \dots & C_{0,m-1} \\ C_{1,0} & C_{1,1} & \dots & C_{1,m-1} \\ \dots & & & \\ C_{n-1,0} & C_{n-1,1} & \dots & C_{n-1,m-1} \end{bmatrix}$$

każde $C_{i,j}$ jest iloczynem skalarnym i -tego wiersza macierzy A przez j -tą kolumnę macierzy B .

²Wykład 10, iis.ans-elblag.pl/~stefan/Dydaktyka/JezForm/Wyklady/10-Wewn/2-potega komórki 0–25.

Uwaga:

Pierwszej osobie, która przyśle mi (na s.sokolowski@ans-elblag.pl) poprawne rozwiązanie tego zadania, przysługują punkty do zaliczenia laboratorium. Liczba tych punktów zależy od oryginalności i elegancji rozwiązania.
