

Języki formalne i metody kompilacji

Laboratorium nr 10

Stefan Sokołowski

Zadanie 1:

Proszę napisać w języku „wewnętrznym” (i przetestować z użyciem symulatora¹) 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;
```

Program nie powinien wprowadzać żadnego ograniczenia na wielkość liczby n . Można założyć (ale nie ma obowiązku), 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 9²).

Zadanie 2:

Proszę napisać w języku „wewnętrznym” (i przetestować) program obliczający sumę danego ciągu liczb. Program nie powinien wprowadzać żadnego ograniczenia na długość ciągu.

Zadanie 3:

Proszę napisać w języku „wewnętrznym” (i przetestować) program obliczający iloczyn skalarny dwóch danych wektorów równej długości:

$$(v_1, v_2, \dots, v_n) \times (w_1, w_2, \dots, w_n) = v_1 \cdot w_1 + v_2 \cdot w_2 + \dots v_n \cdot w_n$$

Program nie powinien wprowadzać żadnego ograniczenia na długość wektorów. Można założyć (ale nie ma obowiązku), że

- zaraz za końcem programu wpisana jest długość n wektorów, komórka na wynik działania programu, oraz inne zmienne pomocnicze,
- dalsze n komórek zawierają współrzędne pierwszego wektora,
- a dalsze n komórek zawierają współrzędne drugiego wektora.

¹ iis.ans-elblag.pl/~stefan/Dydaktyka/JezForm/Wyklady/09-jez-wewn.c

² iis.ans-elblag.pl/~stefan/Dydaktyka/JezForm/Wyklady/09a.pdf

Zadanie 4:

Proszę napisać w języku „wewnętrznym” (i przetestować) program obliczający iloczyn skalarny dwóch przekątnych dwuwymiarowej tablicy kwadratowej, czyli wielkość

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

Program nie powinien wprowadzać żadnego ograniczenia na wielkość tablicy.

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/ćwiczeń. Liczba tych punktów zależy od oryginalności i elegancji rozwiązania. W mailu proszę koniecznie podać swoje **nazwisko**, **numer zajęć** i **numer zadania**.

Zadanie 7 (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 zakończy 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 na dzielenie³ omówionego na wykładzie 10⁴; 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 8 (domowe na punkty):

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

10 → 12	bo	10 = 1 · 8 + 2
19 → 23	bo	19 = 2 · 8 + 3
355 → 543	bo	355 = (5 · 8 + 4) · 8 + 3

Do wyliczania kolejnych cyfr ósemkowych potrzebne jest dzielenie z resztą. Proszę do tego użyć tego samego podprogramu na dzielenie, co w zad. 7.

Uwaga:

Czas na przysłanie (na s.sokolowski@ans-elblag.pl): 2 tygodnie od zajęć 10 — potem już nie przyjmuję rozwiązań. Liczba punktów zależy od oryginalności i elegancji rozwiązania. W mailu proszę koniecznie podać swoje **nazwisko, numer zajęć i numer zadania**.

³ iis.ans-elblag.pl/~stefan/Dydaktyka/JezForm/Wyklady/10-Wewn/2-potega, wiersze 0–25

⁴ iis.ans-elblag.pl/~stefan/Dydaktyka/JezForm/Slajdy/10a.pdf