

3. zadaća - Uvod u računarstvo

1. Napišite program koji učitava $n \in \mathbb{N}$ te polje $n + 1$ prirodnih brojeva koji su koeficijenti polinoma $p(x) = \sum_{i=0}^n a_i x^i$. Program treba računati:

- $$S1 = p\left(\prod_{i=0}^n a_i\right) - p\left(\sum_{i=0}^n a_i\right)$$

- $$S2 = p(A) + p(B)$$

pri čemu je A najveći prosti faktor najvećeg koeficijenta polinoma, a B najveći koeficijent po apsolutnoj vrijednosti, odnosno $B = \max_i a_i$

Napomena 1: Vrijednosti polinoma u točki treba računati Hornerovim algoritmom, funkcijom.

Napomena 2: Pretpostavka je da traženi najveći prosti faktor postoji.

2. Napisati program koji učitava prirodan broj n , te polje a prirodnih brojeva od n elemenata. Napisati funkcije `sort1` i `sort2` koje za argument uzimaju polje te broj njegovih elemenata, a vraćaju sortirano polje. Funkcije `sort1` i `sort2` trebaju koristiti različite algoritme sortiranja.
3. Napisati program koji učitava prirodne brojeve n i $b \geq 2$, te polje x prirodnih brojeva od n elemenata. Program treba konstruirati polje y , kojemu su elementi redom sume znamenaka elemenata polja x , u sustavu sa bazom b . Program treba pronaći sumu kvadrata znamenaka u sustavu sa bazom $b+1$ aritmetičke sredine najmanjeg i najvećeg elementa polja y .

Dodatni zadatak-NEOBVEZAN

Za rješavanje zadatka potrebno je znati Turingov stroj, brojevne sustave, Hornerov algoritam(poželjno ali ne nužno), osnove struktura podataka, te osnove jezika C.

4. Napišite program koji simulira rad Turingovog stroja koji za broj na traci zapisan u sustavu sa bazom 3, pronalazi ostatak pri dijeljenju sa $(7)_{10}$. Pod simulacijom rada stroja podrazumijeva se generiranje konačnog niza petorki oblika (q_i, s_j, q_k, s_l, m) (pogledati definiciju sa predavanja ili vježbi) koji u potpunosti opisuju rad stroja.

Smjernice pri rješavanju:

Koristeći Hornerov algoritam, konstruirati funkciju prijelaza danog Turingovog stroja (dovoljno je 7 stanja). Osmisliti strukturu podataka koja će biti najprikladnija za C implementaciju dotičnog Turingovog stroja. U kontekstu definirane i implementirane funkcije prijelaza, Turingov stroj za dani input (sadržaj trake) generira niz petorki.

Napomena:

Implementacija navedenog problema nije jedinstvena. Zadatak je predviđen za one studente koji se žele iskušati u samostalnom (bez pomoći asistenata i demonstratora) rješavanju problema koji težinom malo iskaču van predviđenih nastavnih protokola kolegija.