

Objektno programiranje

vježba 3

uvod u stl

1. Napisati funkciju koja će sumirati prvi i zadnji element vektora, drugi i predzadnji, treći i predpredzadnji itd. Izračunate sume spremite u novi vektor te izračunati najveću i najmanju među sumama.
2. Napisati funkciju koja izmiješa riječi u rečenici u inverznom poredku. Točka mora ostati na kraju rečenice, prva riječ mora biti napisana velikim početnim slovom, a zadnja malim. Koristiti reverse funkciju.
3. Napisati funkciju koja iz stringa izbaci svako pojavljivanje podstringa koristeći funkcije standardne biblioteke.
4. Napisati funkciju koja vraća najduži zajednički podstring u dva stringa.
5. Napisati funkciju koji prevodi engleske rečenice na pig latin jezik. Pravila su sljedeća:

- (a) ako riječ počinje samoglasnikom dopisuje se hay na kraj riječi,
- (b) inače, svi suglasnici s početka riječi prebacuju se na kraj te se na kraju riječi dopisuje ay.

Rečenice treba spremite u vector, a zatim prevesti slučajno odabranu rečenicu iz vektora.

Primjer: "What time is it?" prevodi se kao "atwhay imetay ishay ithay?"

6. Napisati funkciju koja u listu prirodnih brojeva ispred svakog parnog broja umetne nulu, a iza svakog neparnog broja osim jedinice umetne jedinicu. Funkcija vraća broj elemenata između prve nule u listi i zadnje jedinice. U main funkciji generirati listu, pozvati funkciju, ispisati listu i povratnu vrijednost funkcije.
7. Napisati program u kojem korisnik unosi binarne znamenke (0 i 1) i sprema ih u container. Unosu je kraj kad korisnik unese vrijednost koja ne predstavlja binarnu znamenku. Nakon toga napisati funkciju stuffing_bit koja primjeni bit stuffing algoritam na container. Bit stuffing je tehnika koja se koristi u HDLC protokolima za prijenos podataka u kojima je početak i završetak paketa određen sa nizom bitova 01111110. Da se ne bi sadržaj paketa brkao sa stop-bitovima, potrebno je nakon svakog niza od pet jedinica umetnuti jednu nulu.

Primjer: za uneseni niz bitova

0011101111111000111111011

rezultat je

0011101111101100011111011011.

8. Napisati funkciju `unstuffing_bit` koja je inverzna funkciji `stuffin_bit` iz predhodnog zadatka. Funkcija prima modificirani container i izbacuje umetnute nule kako bi se dobila originalna sekvenca.