

Basic of Informatics – Lab. 10. //structures, typedefs, input/output

1. Define enum-type variable for storage the marks (from ndst to bdb) and use this variable to control the switch statement that will be displaying the mark depending on average from partial marks. Define another switch statement with case ranges (...).
2. Using strcpy, strlen, strncpy, strcat, strncat, strcmp, toupper and tolower functions, write a program to enter the names of several users, and then allow you to paste all the names and only a few letters from individual names into one string. The program should enter the same name once. In one case, print the resulting string as lower case letters and then as uppercase letters.
3. Define structure Person with fields (with correct type): Name, Age, Height, Sex (W/M). Define another structure Person1, with the same fields, but with initializing and p3 with initializing with designators. Create a program with 3 variables (structures): p1, p2 and p3. Then set up fields by assignments structure p1. Then create a pointer to variable p1. Display values of p1 (address and the fields) with use of pointers and by referring to individual fields (like p1.Name).
4. Using structure p1 from pervious exercise declare an array of students and initialize it by names of your colleagues . Fill another fields with random data and display all this data on the screen (with any loop). Try different way to initialize values of structure p1 (with designators).
- 4*. Modify structure Person and define a pointer to structure Person itself and implement a linked list with two positions – use example from lectures.
5. Define a function person1 returning type struct Person (from ex. 3): struct Person and with formal arguments the same as in the structure Person (fields Name, Age, Height, Sex). The function of person1 is to display complete information about the person on the screen (like: My name is: /Name/. I'm /Age/, etc.
6. Define a function that will be comparing age of data of people included in the structure Person Arguments of this function should be type of structure (Person). Also You can compare a length of names.
7. Use typedef declaration to simplifying declaration of structures //typedef struct {...}Person;
8. Create a file: data.txt in the same directory as *.exe file. In file data.txt put some sentences with small and capitol letters. Then define two programs: one that will be using getchar and putchar to convert small letters to capital letters – keyboard and second program that will be converting letters from file data.txt. //display on the screen and save in output file.
9. Using fprintf and fscanf modify yours quadratic function – as a, b, c -coefficients use data from text file (check correctness of loading all data).