Chapters
1 2 3 4
5 6 7 8
9 10 11 12
13

Chapter 13
- Theory (9)
- Programming exercises (2/2)
- Quizzes (4/4)
- Open exercises

Course
- Table of contents
- Extra materials
- Bulletin board
- My corner

Communication
- Forum
- Conference

Tutoring
- Ask a tutor

Table processing and pointers

Feedback | Ask a tutor
Please use this option only if you find any issues or mistakes in the content. If you have any queries or questions regarding the subject matter, then please use "Ask a tutor"

Page: 2/2

- (1) Returning a pointer from a function done
- (2) Table processing and pointers done

You have completed this exercise

Send

Hide

The C program below receives integers as command line arguments. After receiving the required values, the program calls the subroutine "calculate_and_print" which receives an array and its size passed by the main program, prints the numbers on the screen and prints their sum on the screen.

```
#include <stdio.h>
#include <stdlib.h>

void calculate_and_print(int *, int);

int main(int argc, char *argv[])
{
  int x, sum=0, size=5, array[5];
  if(argc == 6){
  /* Program name and command line parameters */
    for(x=0;x<argc-1;x++){
     array[x] = atoi(argv[x+1]);
    }
    calculate_and_print(array, size);
  }else{
    printf("Incorrect number of command line arguments\n");
  }
  return 0;
}
```

Your task is to write the subroutine "calculate_and_print". The example printout shows how to print the numbers. The main program includes things that are outside the scope of this course, but you only need to write a subroutine that serves the main program.

Example output:

```
Elements of the array: 43 53 654 65 77
Sum = 892
```

The verification of program output does not account for whitespace characters like "\n", "\t" and " "

- program.c

```c
#include <stdio.h>
#include <stdlib.h>

void calculate_and_print(int *, int);

int main(int argc, char *argv[])
{
  int x, sum=0, size=5, array[5];
  if(argc == 6){
  /* Program name and command line parameters */
    for(x=0;x<argc-1;x++){
      array[x] = atoi(argv[x+1]);
    }
    calculate_and_print(array, size);
  }else{
    printf("Incorrect number of command line arguments\n");
  }
  return 0;
}
void calculate_and_print(int *array, int size)
{
int sum=0;
for (int i = 0; i < 5; i++)
    sum += array[i];

printf("Elements of the array: %d %d %d %d %d\n", array[0], array[1], array[2], array[3], array[4], array[5]);
printf("Sum = %d\n", sum);
}
```

Position: Ln 20 Ch 1 Full screen (Esc to exit)

Reset Save Run