Ceng 198 Introduction to Programming
Final
May 28, 2008 09.00 – 10.50
Good Luck!

1. (40 pts) Write a complete C program that makes a call to a function that will take an array of size 250 as an argument.

   - The array should be filled with random numbers from 1-100. The filling of the array will be done within the *same* function.
   - The print out of the numbers with their indices will also be done in the function.
   - The function shoul also find/calculate and then print out the following properties of the array;
     - the maximum and corresponding index,
     - the minimum and corresponding index,
     - the arithmetic mean (average, $\overline{x}$),
     - the standard deviation $\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})^2}$, where x is the name of the array, i is the array index and N is the size of the array (250).
   - The function will not return anything.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
#define SIZE 250
void myfunction(int arrayinfunction[], int size);
int main()
{
  int arrayinmain[SIZE];
  myfunction(arrayinmain, SIZE);
  return 0;
}
void myfunction(int arrayinfunction[], int size)
{
  int i, min, max, minIndex, maxIndex;
  float average, sum, sigma;
  srand(time(0));
  for (i = 0; i < size; i++)
```

```
    arrayinfunction[i] =1+rand()%100;
  for (i = 0; i < size; i++)
    printf("number=%d index=%d \n",arrayinfunction[i],i);
  /*Maximum and corresponding index*/
  max = arrayinfunction[0];
  for (i = 1; i < size; i++)
    if (arrayinfunction[i] > max)
      {
max = arrayinfunction[i];
maxIndex = i;
      }
  printf("Max. Number: %d, Index of Max. Number: %d\n", max, maxIndex);
  /*Minimum and corresponding index*/
  min = arrayinfunction[0];
  for (i = 1; i < size; i++)
    if (arrayinfunction[i] < min)
      {
min = arrayinfunction[i];
minIndex = i;
      }
  printf("Min. Number: %d, Index of Min. Number: %d\n", min, minIndex);
  /*Arithmetic Mean*/
  sum=0.0;
  for (i = 0; i < size; i++)
    sum=sum+arrayinfunction[i];
  average=sum/size;
  printf("Arithmetic mean (average) of %d numbers is %f \n", size, average);
  /*Standart Deviation*/
  sum=0.0;
  for (i = 0; i < size; i++)
    sum=sum+(arrayinfunction[i]-average)*(arrayinfunction[i]-average);
  sigma=sqrt(sum/size);
  printf("Standart deviation is %f \n", sigma);
  return;
}
```

2. (25 pts) Define a function with the name "'Calculator"

   - that takes 3 parameters; first operand with data type float, second operand with data tpye float and the operator with data type chratacter.
   - The function should make a calculation according to the given character and then return the value calculated to main.
   - These three parameters and resulting calculated value should be printed out in tne main function.

```
#include <stdio.h>
float simpleCalculator(float, char, float);
int main()
{
  float firstOp, secondOp,result;
  char oper;
  printf("Enter operator (+,-,*, or /): ");
  scanf("%c", &oper);
  printf("Enter first operand: ");
  scanf("%f", &firstOp);
  printf("Enter second operand: ");
  scanf("%f", &secondOp);
  result = simpleCalculator(firstOp, oper, secondOp);
  printf("Result of the operation %f %c %f is %f\n", firstOp, oper, secondOp,
          result);
  return 0;
}
float simpleCalculator(float fo, char op, float so)
{
  if(op == '+')
    return fo + so;
  else if(op == '-')
    return fo - so;
  else if(op == '*')
    return fo * so;
  return fo / so;
}
```

3. (25 pts) Write a complete C program that makes a call to a function

   - that takes three arguments: a character and two integers.
   - The character is to be printed as
     - the first integer specifies the number of times that the character is to be printed on a line,
     - the second integer specifies the number of lines that are to be printed.

```
#include <stdio.h>
void printcharacter();
int main()
{
  printcharacter();
  return 0;
}
void printcharacter()
{
  int i,j,integer1, integer2;
  char character;
  printf("Enter character : ");
  scanf("%c", &character);
  printf("Enter first integer: ");
  scanf("%d", &integer1);
  printf("Enter second integer: ");
  scanf("%d", &integer2);
  for(i=1;i<=integer2;i++)
    {
      for(j=1;j<=integer1;j++)
printf("%c",character);
      printf("\n");
    }
  return;
}
```

4. (20 pts) A palindrome is a number or a text phrase that reads the same backwards as forwards. For example, each of the following five–digit integers is a palindrome: 12321, 55555, 45554 and 11611. Write a complete C program that reads in a five–digit integer and determines whether it is a palindrome.
   Hints:

   - Use the division and modulus operators to seperate the number into its individual digits.
   - Store each digit in its own variable.

```
#include<stdio.h>
int main(void)
{
  int i,number, onlar=10,a,b,c,d,e;
  printf("Enter a five-digit integer : ");
  scanf(" %d",&number);
  //   printf("number = %d \n",number);
  a=number%onlar;
  number=number/onlar;
  b=number%onlar;
  number=number/onlar;
  c=number%onlar;
  number=number/onlar;
  d=number%onlar;
  number=number/onlar;
  e=number%onlar;
  printf("a=%d b=%d c=%d d=%d e=%d \n",a,b,c,d,e);
  if(a==e && b==d)
    printf("Entered number %d%d%d%d%d is a palindrome\n",e,d,c,b,a);
  else
    printf("Entered number %d%d%d%d%d is NOT a palindrome\n",e,d,c,b,a);
  return 0;
}
```