Modélisation et vérification des systèmes informatiques

Dominique Méry

22 novembre 2015

Table des matières

1 Vérification de programmes écrits en C

Exercice 1 *Vérifier le programme C suivant en la traduisant dans un module PlusCal.*

```
    /* C Program to find roots of a quadratic equation when coefficients are entere
/* Library function sqrt() computes the square root. */

#include <stdio.h>
#include <math.h> /* This is needed to use sqrt() function.*/
int main()
{
  float a, b, c, determinant, r1,r2, real, imag;
  printf("Enter coefficients a, b and c: ");
  scanf("%f%f%f",&a,&b,&c);
  determinant=b*b-4*a*c;
  if (determinant>0)
  {
      r1= (-b+sqrt(determinant))/(2*a);
      r2= (-b-sqrt(determinant))/(2*a);
      printf("Roots are: %.2f and %.2f",r1 , r2);
  }
  else if (determinant==0)
  {
    r1 = r2 = -b/(2*a);
    printf("Roots are: %.2f and %.2f", r1, r2);
  }
  else
  {
    real= -b/(2*a);
    imag = sqrt(-determinant)/(2*a);
    printf("Roots are: %.2f+%.2fi and %.2f-%.2fi", real, imag, real, imag);
  }
  return 0;
}
```

Output 1

```
Enter coefficients a, b and c: 2.3
4
5.6
Roots are: -0.87+1.30i and -0.87-1.30i
```

Output 2

```
Enter coefficients a, b and c: 4
1
0
Roots are: 0.00 and -0.25
```

*To solve this program, library function sqrt() is used. This function calculates th
sqrt() function.*
Similar C Programming Examples
C Program to Add Two Complex Numbers by Passing Structure to a Function
C Program to Find Quotient and Remainder of Two Integers Entered by User

C Program to Check Prime or Armstrong Number Using User–defined Function
C Program to Display Prime Numbers Between Intervals Using User–defined Function
C Program to Display its own Source Code as Output

Exercice 2 *Vérifier le programme C suivant en la traduisant dans un module PlusCal.*

```
/* C program to check whether a number is palindrome or not */

#include <stdio.h>
int main()
{
  int n, reverse=0, rem,temp;
  printf("Enter an integer: ");
  scanf("%d", &n);
  temp=n;
  while(temp!=0)
  {
     rem=temp%10;
     reverse=reverse*10+rem;
     temp/=10;
  }
/* Checking if number entered by user and it's reverse number is equal. */
  if(reverse==n)
      printf("%d is a palindrome.",n);
  else
      printf("%d is not a palindrome.",n);
  return 0;
}
```

Exercice 3 *Vérifier le programme C suivant en la traduisant dans un module PlusCal.*

```
    #include <stdio.h>
int main()
{
  int n,count=0;
  printf("Enter an integer: ");
  scanf("%d", &n);
  while(n!=0)
  {
      n/=10;                /* n=n/10 */
      ++count;
  }
  printf("Number of digits: %d",count);
}
```

Exercice 4 *Vérifier le programme C suivant en la traduisant dans un module PlusCal.*

```
     #include<stdio.h>
int main(){
    char line[150];
    int i,v,c,ch,d,s,o;
    o=v=c=ch=d=s=0;
    printf("Enter a line of string:\n");
    gets(line);
    for(i=0;line[i]!='\0';++i)
    {
```

```
        if(line[i]=='a' || line[i]=='e' || line[i]=='i' || line[i]=='o' || line[i]==
            ++v;
        else if((line[i]>='a'&& line[i]<='z') || (line[i]>='A'&& line[i]<='Z'))
            ++c;
        else if(line[i]>='0'&&c<='9')
            ++d;
        else if (line[i]==' ')
            ++s;
    }
    printf("Vowels: %d",v);
    printf("\nConsonants: %d",c);
    printf("\nDigits: %d",d);
    printf("\nWhite spaces: %d",s);
    return 0;
}
```

Exercice 5 *Vérifier le programme C suivant en la traduisant dans un module PlusCal.*

```
    #include<stdio.h>
int main(){
    char line[150];
    int i,j;
    printf("Enter a string: ");
    gets(line);
    for(i=0; line[i]!='\0'; ++i)
    {
        while (!((line[i]>='a'&&line[i]<='z') || (line[i]>='A'&&line[i]<='Z' || lin
        {
            for(j=i;line[j]!='\0';++j)
            {
                line[j]=line[j+1];
            }
            line[j]='\0';
        }
    }
    printf("Output String: ");
    puts(line);
    return 0;
}
```

Exercice 6 *Vérifier le programme C suivant en la traduisant dans un module PlusCal.*

```
    #include <stdio.h>
int main()
{
  int n, reverse=0, rem;
  printf("Enter an integer: ");
  scanf("%d", &n);
  while(n!=0)
  {
     rem=n%10;
     reverse=reverse*10+rem;
     n/=10;
  }
  printf("Reversed Number = %d",reverse);
  return 0;
}
```