Gabarito da Prova 2
MAC115 - Física, segundo semestre de 2000

1. A saída é como segue:

```
Os elementos da sequencia sao: 3  6  1  4  7  8  5
0: 9
1: 7
2: 5
3: 11
4: 15
5: 13
A sequencia dada e' alternante.
[1: 3 6]
[2: 1 6]
[3: 1 6]
[4: 1 7]
[5: 1 8]
[6: 1 8]
Menor = 1   Maior = 8
```

2. (*a*)

```
int eh_crescente(int v[], int n)
{
  int j;

  for (j=1; j<n; j++)
    if (v[j]<=v[j-1])
      return 0;

  return 1;
}
```

(*b*)

```
int eh_K_crescente(int a[][NMAX], int m, int n, int K)
{
```

```
  int i;

  for (i=0; i<m; i++) {
    if (!eh_crescente(a[i], n))
      return 0;
    if ((a[i][0]<=0)||(a[i][n-1]>K))
      return 0;
  }
  return 1;
}
```

(c)

```
#include <stdio.h>

#define MMAX 100
#define NMAX 100

int eh_crescente(int v[], int n);
int eh_K_crescente(int a[][NMAX], int m, int n, int K);

int main()
{
  int m, n, i, j, K, max_el=0;
  int M[MMAX][NMAX];

  scanf("%d %d", &m, &n);

  for (i=0; i<m; i++)
    for (j=0; j<n; j++)
      scanf("%d", &M[i][j]);

  scanf("%d", &K);

  if (!eh_K_crescente(M, m, n, K))
    printf("Matriz nao eh %d-crescente.\n", K);
  else {
    printf("Matriz eh %d-crescente.\n", K);

    for (i=0; i<m; i++)
      if (M[i][n-1]>max_el)
        max_el=M[i][n-1];
```

```
    printf("Matriz eh tambem %d-crescente, mas nao eh %d-crescente.\n",
           max_el, max_el-1);
  }

  return 0;
}
```

3. (*a*)
```
void leia_pol(double p[], int n)
{
  int i;

  for (i=0; i<=n; i++)
    scanf("%lf", &p[i]);
}
```

(*b*)
```
void imprima_pol(double p[], int n)
{
  int i;

  printf("grau: %d\n", n);
  printf("coeficientes: ");
  for (i=0; i<=n; i++)
    printf("%g ", p[i]);
  printf("\n");
}
```

(*c*)
```
double valor(double p[], int n, double x)
{
  int i;
  double xx=1., y;

  y=p[0];
  for (i=1; i<=n; i++) {
    xx *= x;
    y += p[i]*xx;
  }

  return y;
}
```

(*d*)
```
#include <stdio.h>
```

```c
#define NMAX 1000

void leia_pol(double p[], int n);
void imprima_pol(double p[], int n);
double valor(double p[], int n, double x);

int main()
{
  int n, m, i;
  double p[NMAX], x;

  scanf("%d", &n);
  leia_pol(p, n);
  imprima_pol(p, n);

  scanf("%d", &m);
  for (i=0; i<m; i++) {
    scanf("%lf", &x);
    printf("p(%g) = %g\n", x, valor(p, n, x));
  }

  return 0;
}
```