What is the output of the C program shown below?

```
int g(int n) {
    for (int i=0; i<10; i++) {
        if (i*i == n)  return i;
    }
    return 0;
}
int f(int n) {
    if (n == 0) return 0;
    return g(n%10) + f(n/10);
}
int main(void) {
    printf("%d\n", f(4) );
    printf("%d\n", f(25) );
    printf("%d\n", f(165) );
    printf("%d\n", f(1024) );
    printf("%d\n", f(79146) );
    return 0; }
```

Give the output of the C program below when run with **`./a.out ABCD B7 Hello F18`**

```
int *f(int n, char str[]) {
    int *p = malloc(sizeof(int)*n);
    int len=strlen(str);
    for (int i=0; i<n; i++) {
        p[i] = 0;
        if (i<len) {
            if (isupper(str[i]))
                p[i] = str[i]-'A';
            else if (islower(str[i]))
                p[i] = str[i]-'a';
            else if (isdigit(str[i]))
                p[i] = str[i]-'0';
        }
    }
    return p;
}
int g(int *x, int n) {
    int m=0;
    for (int i=0; i<n; i++)
        if (x[i]>m) m=x[i];
    return m;
}
int main(int argc, char *argv[]) {
    int n=argc-1;
    int **a=malloc(sizeof(int*)*n);
    for (int i=0; i<n; i++)
        a[i]=f(n, argv[i+1]);
    int *p = malloc(sizeof(int)*n);
    for (int i=0; i<n; i++) {
        p[i]=g(a[i], n);
        printf("%d: %d\n", i, p[i]);
    }
    printf("G: %d\n", g(p, n));
    return 0; }
```

Give the output of the C program shown below when the user enters the following: **4 7 8 5 3 6 9 2 0**

```
#include <stdio.h>
#include <stdlib.h>
typedef struct _node {
    int data;
    struct _node *next;
} Node;
Node *a(int num, Node *ptr) {
    Node *pTemp = malloc( sizeof(Node) );
    pTemp->data = num;
    pTemp->next = ptr;
    return pTemp;
}
Node *b(Node *ptr1, Node *ptr2) {
    if (ptr1==NULL) return ptr2;
    ptr1->next=b(ptr1->next, ptr2);
    return ptr1;
}
void c(Node *ptr) {
    printf("[ ");
    for (Node *cur = ptr; cur != NULL; cur = cur->next)
        printf("%d ", cur->data);
    printf("]\n");
}
int main(void) {
    Node *ptr1 = NULL;
    Node *ptr2 = NULL;
    while (1) {
        int value;
        scanf("%d", &value);
        if (value==0) break;
        Node *ptr=a(value, NULL);
        if (value%2!=0)
            ptr1=b(ptr1, ptr);
        else
            ptr2=b(ptr2, ptr);
    }
    c(ptr1);
    c(ptr2);
    b(ptr1, ptr2);
    c(ptr1);
    c(ptr2);
    return 0;
}
```

Please note the #include directives are omitted in some programs to conserve space.