

1. Find the bug in the following function:

```
double power(double x, int n)
{
    if (n<0)
        return power(1.0/x, -n) ;
    else switch (n)
        {
        case 0: return 1.0 ;
        case 1: return x ;
        default:
            return power(power(x,n/2),2)
                * power(x,n % 2) ;
        }
}
```

2. (a) How do you find the space used by a recursive program?  
(b) How do you find the time used by a recursive program?
3. (a) What is tail-call optimization? Can it apply to a non-recursive function?  
(b) Is the following function tail-recursive?

```
// Compute x ** n * a
double power2(double x, int n, double a)
{
    if (n<0)
        return power2(1.0/x, -n, a) ;
    else switch (n)
        {
        case 0: return a ;
        case 1: return x*a ;
        default:
            return power2(x*x, n/2, power2(x, n%2, a)) ;
        }
}
```

- (c) Is it correct?