In theory of probability, the beta distribution function is given by

$$f(x;p,q) = \frac{1}{B(p,q)} x^{p-1}(1-x)^{q-1} \qquad 0 \le x \le 1; \quad p,q > 0$$

where $B(p,q) = \int_0^1 t^{p-1}(1-t)^{q-1}\mathrm{d}t$, is the beta function.

(a) Write a C or C++ function `double betadist(double x, double p, double q)` that evaluates the beta distribution function using the above two definitions. You may use any numerical integration method of your choice. [5]

```
#include <iostream>
#include <cstdlib>
#include <cmath>
#include <fstream>

using namespace std;

double f(double t, double p, double q)
{
   return pow(t, p-1) * pow(1-t, q-1);
}

double simpson13(double p, double q)
{
   int N = 50;
   double a = 0;
   double b = 1;
```

```
    double h = (b - a) / (double) N;

    double sum = 0;
    for(int i=1; i<=N-1; i+=2) {
        sum += 4 * f(a+i*h, p, q);
    }
    for(int i=2; i<=N-2; i+=2) {
        sum += 2 * f(a+i*h, p, q);
    }

    return (h/3.0) * (f(a, p, q) + sum + f(b, p, q));
}

double betadist(double x, double p, double q)
{
    return pow(x, p-1) * pow(1-x, q-1) / simpson13(p, q);
}
```