CS601: Software Development for Scientific Computing

Final Exam, 15/11/2022, 2:30PM-5:30PM, Max points: 35

Instructions : this exam is open book, open notes (printed/written). No electronic devices allowed. Question 10 (last) is take home and is optional. The instructions for this question are given separately in the README file provided along with the GitHub repo. Write your answers legibly. State your assumptions (if any) clearly.

1. **[4 points]** Consider the macro: `#define MAX((a),(b)) ((a)>(b)?(a):(b))`

   a Show with an example code snippet, a major disadvantage of using this macro. Hint: think when you have function calls (that modify state) as arguments of this macro. [2 points]

   b Propose an alternative solution when you know that the arguments are only of type `double`. You must use the `inline` keyword in your proposed solution. [1 point]

   c Propose another 'generic' solution that works for several other types. You must use the `const` and `inline` keywords in your proposed solution. [1 point]

2. **[5 points]** Consider part of the program shown below and a gdb session in progress (truncated pic of gdb session due to lack of space) to analyze the segmentation fault that happens *sometimes* with the program.

```
7  short generate_randshort(){
8          short sum=1;
9          for(short i=1;i<(rand() % N);++i) {
10                 for(short j=1;j<N;++j) {
11                         int val=(i*j)/(i+j);
12                         sum += (val!=0)?val:sum;
13                 }
14         }
15         return sum;
16 }
17
18 long long fact(unsigned int x){
19         if(x==1 || x==0)
20                 return 1LL;
21         else
22                 return x*fact(x-1);
23 }
24
25 int main(int argc, char* argv[]){
26         short f1=0;
27         //provide a seed to srand based on process id
28         srand((unsigned int)getpid());
29         //generate a random number
30         f1=generate_randshort() % 10;
31         //get the factorial
32         long long factrandom=fact(f1);
33         //normalize
```

```
Program received signal SIGSEGV, Segmentation fault.
0x00005555555552c5 in fact (
    x=<error reading variable: Cannot access memory at address 0x7fffff7fefec>)
    at gdbdemo.cpp:18
18      long long fact(unsigned int x){
(gdb) bt
#0  0x00005555555552c5 in fact (
    x=<error reading variable: Cannot access memory at address 0x7fffff7fefec>)
    at gdbdemo.cpp:18
#1        part a         in fact (x=  part b    ) at gdbdemo.cpp:part f
#2        part a         in fact (x=  part c    ) at gdbdemo.cpp:part f
#3        part a         in fact (x=  part d    ) at gdbdemo.cpp:part f
#4        part a         in fact (x=  part e    ) at gdbdemo.cpp:part f
#5  0x00005555555552f0 in fact (x=4294792595) at gdbdemo.cpp:22
```

   a What is a prerequisite to see meaningful information via the `gdb` debug session? What does `gdb a.out` followed by `r` commands do? (Assume name of the executable is `a.out` and the executable does not accept any command line arguments.) [1 point]

   b what are the values of `part a`, `part b`, `part c`, and `part f`? [2 points]

   c Why does the program crash sometimes and how do you fix the bug? [2 points]

3. **[8 points]** consider the following set of points in 2D space: `(5.5, 6.5)`, `(1.5, 3.5)`, `(0.5, 2.5)`, `(0.5, 0.5)`, `(2.5, 0.5)`, `(3.5, 0.5)`,`(6.5, 0.5)`, `(6.5, 1.5)`, and `(4.5, 3.5)` (a point `(x, y)` denotes `x` units along `X`-axis and `y` units along `Y`-axis).

   a Draw an adaptive quad-tree representing the space of points. Assume that the leaf nodes can have at most one point. [2.25 points]

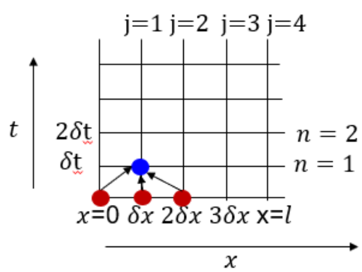
   b Show a numbering of the leaf nodes as per z-order numbering. [2.25 points]

   c Retrieve all points in the range `(4,4)` to `(6,1)`. If you did this retrieval naïvely for N-points, each in K-Dimensional space, what would be the time complexity? [1 point]

   d Using the quad-tree constructed, propose a scheme to retrieve the set of points (for the range mentioned previously) efficiently. You must write a pseudocode or code snippet. You may assume that a member method `bool DoesIntersect(QuadTreeNode* node, Box* b)` of class `QuadTreeNode` is given. Class `Box` represents the bounding box (a rectangle in this space). Class `Point` represents a point. [2.5 points]

4. **[2 points]** Classify the following problems as regular/irregular w.r.t. the computing and communication pattern involved: a) Barnes-Hut b) Delauney Mesh Refinement c) Minimum Weight Triangulation d) Sparse-Matrix vector multiplication

5. **[2 points]** The picture below shows a snapshot of the stencil computation for the heat diffusion problem with appropriate initial and boundary conditions given. The dot at `(j=1, n=1)` indicates *part of the computation* to be done at time step 1 (i.e. at time `dt` from the beginning), using the data values from the initial conditions (shown using dots with arrows pointing to `(j=1, n=1)`).



   a Is this a 1D problem or 2D problem? Is this 4-point, 3-point, or 2-point stencil computation?

   b Name the independent variables. Mention one disadvantage of the 3-point stencil computation?

6. **[3 points]** Consider the *total amount* of work involved involved in *any single time step* in 3-point stencil computation and Barnes-Hut (force calculation stage only).

   a Devising a strategy to distribute the work involved among multiple workers/cores of a system is easy in both problems. True or False? Why?

   b In which of the problems you can exploit locality better? Why?

7. **[5 Points]** Consider a second order differential equation

$$\frac{d^2u}{dx^2} = u + x \tag{1}$$

subjected to

$$u(0) = 0, \qquad u(1) = 0$$

The domain of the problem is [0,1]. Equation (1) is discretized using finite difference method using 1, 2, 3, 4 and 5 grid points as shown in Fig. where, 1 and 5 grid points denote boundary points.

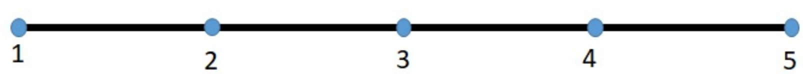


Figure 1: Figure for Question-7

Answer the following.

- Write down the discrete equations for grid point 2, 3 and 4

- Consider step size $(h) = 0.25$ and rewrite the equations.
- Apply the boundary conditions and represent equations in the matrix format, $Ax = b$ form.
- What is the type of matrix A and what is its bandwidth.
- Compute value of u at grid points 2,3 and 4.

8. **[3 points]** Write the weak form of following second order differential equation.

$$-\frac{d^2u}{dx^2} - u + x^2 = 0, \qquad 0 < x < 1 \tag{2}$$

subjected to,

$$u(0) = 0, \qquad u(1) = 0$$

Identify the primary and secondary variables from the weak form.

9. **[3 points]** The elemental stiffness matrix for the 2 noded one dimensional element is given by, k= $\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$. The domain is discretized using one dimensional 4 elements as shown in Fig. 2. Write the global stiffness

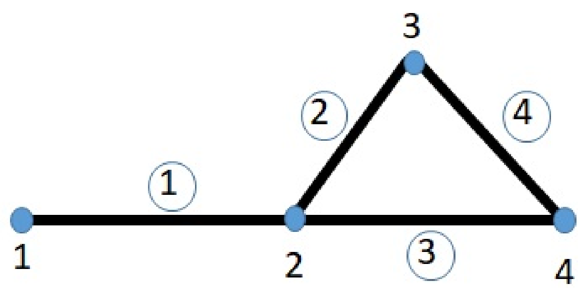


Figure 2: Figure for Question-9

matrix. [Hint: You need to assemble all the elemental matrix. The size of global stiffness matrix will be 4×4]

10. **[Optional: 2 points]** Write a program to evaluate the integral given in Eq. (3) using 4 point Gauss quadrature rule

$$I = \int_{-1}^{+1} \cos\frac{\pi x}{2} dx \tag{3}$$

The weights and sampling points are given as,

$$[w_1, w_3 = 0.347854845], [w_2, w_4 = 0.652145155],$$

$$\xi_1 = -0.3399810435, \xi_3 = 0.3399810435, \xi_2 = -0.8611363116, \xi_2 = 0.8611363116$$

.

What is the percentage absolute error between numerically computed value of integral and actual value?

to answer this question, you must click on the link shared with you in the discussion forum to create a Github repo and follow the instructions mentioned in README file present in the repo.