Physics 40: Final Examination

Friday, June 7, 2019

Examination is closed book. No cell phones or calculators.

The final has four parts:
Part One: asks some short coding questions, similar to your quizzes.
Part Two: contains some general questions about using computers for computational physics.
Part Three: a few questions about mathematics and physics from this course.
Part Four: writing complete codes from scratch.

PART ONE

[1.] In the space provided, complete the C code needed to print the first 20 odd integers (using a loop).

```
#include <stdio.h>
#include <math.h>
int main(void)
{
int j;
                                                          <
                                                          <
                                                          <
                                                          <
                                                          < YOUR CODE
                                                          <    HERE
                                                          <
                                                          <
                                                          <
return 0;
}
```

[2.] In the space provided, complete the C code needed to decide whether the integer j is even or odd.

```
#include <stdio.h>
#include <math.h>
int main(void)
{
int j;
printf("Enter the integer j\n");
scanf("%i",&j);
                                                          <
                                                          <
                                                          <
                                                          <
                                                          < YOUR CODE
                                                          <    HERE
                                                          <
                                                          <
                                                          <
return 0;
}
```

[3.] Use a C do-while loop to write a code to count down from 11 to 0, that is, to print 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0 to the screen.

```
#include <stdio.h>
#include <math.h>
int main(void)
{
int j;
                                                    <
                                                    <
                                                    <
                                                    < YOUR CODE
                                                    <    HERE
                                                    <
                                                    <
return 0;
}
```

[4.] The python script below initializes a vector (array) to $V = (0.4, 0.0, 0.6, 0.5, 0.5)$. Complete it so that it computes, and prints, the participation ratio $\mathcal{P}$ of V.

```
# THIS IS NEEDED TO USE ARRAYS IN PYTHON:
from array import *

# FILL UP THE VECTOR V:
V =array('f',[0.4])
V.append(0.0)
V.append(0.6)
V.append(0.5)
V.append(0.5)
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <  YOUR
                                                    <  CODE
                                                    <  HERE
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
```

From what you know about $\mathcal{P}$, roughly what value do you expect? Explain.

[5.] Write a piece of C code which initializes all 2000 elements of an array so that its values are 1.99 except rho[874]=0.3 and rho[1135]=0.7.

```
#include <stdio.h>
#include <math.h>
int main(void)
{
int i;
double rho[2000];
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <  YOUR
                                                    <  CODE
                                                    <  HERE
                                                    <
                                                    <
                                                    <
return 0;
}
```

[6.] Write a piece of python script which initializes all 2000 elements of an array so that its values are zero except rho[874]=0.3 and rho[1135]=0.7.

```
# THIS IS NEEDED TO USE ARRAYS IN PYTHON:
from array import *
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <  YOUR
                                                    <  SCRIPT
                                                    <  HERE
                                                    <
                                                    <
                                                    <
                                                    <
```

[7.] What command would you use to evaluate x^{57} in C?
What command would you use to evaluate x^{57} in python?

[8.] What will python tell you in response to these two lines:

n=5
n!=2

[9.] The C code below reads in a vector $\vec{V}$ and a matrix M. Complete it so that it computes and prints $\vec{W} = M\,\vec{V}$.

```
#include <stdio.h>
#include <math.h>

int main(void)
{
int i,j,N=5;
double V[N],M[N][N],W[N];

for (i=0; i<N; i=i+1)
    {
    scanf("%lf",&V[i]);
    }
for (i=0; i<N; i=i+1)
    {
    for (j=0; j<N; j=j+1)
        {
        scanf("%lf",&M[i][j]);
        }
     }
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <  YOUR
                                                    <  CODE
                                                    <  HERE
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
                                                    <
```

[10.] What is the remainder when you divide 113 by 12?
What command would you use to evaluate this in C?
What command would you use to evaluate this in python?

[11.] At left is a program, written in C, that molecular dynamics for a mass m on a spring k, writing the results to a file. In the space to the right, compose the analog python script.

```
#include <stdio.h>
#include <math.h>
int main(void) {
FILE * fileout;
fileout=fopen("myfilename","w");
float x,v,k,m,t,dt;
int j,N;
printf("\nEnter number of steps N");
scanf("%i",&N);
printf("\nEnter time step dt");
scanf("%f",&dt);
printf("\nEnter k");
scanf("%f",&k);
printf("\nEnter m");
scanf("%f",&m);
printf("\nEnter starting position");
scanf("%f",&x);
printf("\nEnter starting velocity");
scanf("%f",&v);
for (j=0; j<N; j=j+1)
   {
   t=dt*j;
   x=x+v*dt;
   v=v-k*x*dt/m;
   fprintf(fileout,"%12.4f %12.4lf \n",t,x);
   }
return 0;
fclose(fileout); }
```

PART TWO

[12.] This code computes N factorial.

```
#include <stdio.h>
#include <math.h>
int main(void) {
int j,N,fact;
printf("Enter N \n");
scanf("%d",&N);
fact=1;
for (j=1; j<N; j=j+1)
   {
   fact=fact*j;
   }
printf("%i \n",fact); }
```

What is (roughly) the largest integer N for which it will work? Explain your answer!

[13.] In base ten, the first seventeen non-negative integers are:
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16.
Write these numbers in base two.

[14.] This C code does Ntimesteps=10^5 MD steps for Nparticles=10^7 masses $m[\,j\,]$ on springs $k[\,j\,]$:

```
Ntimesteps=pow(10,5);
Nparticless=pow(10,7);
for (t=0;t<Ntimesteps;t=t+1)
   {
   for (j=0;j<Nparticles;j=j+1)
      {
      x[j]=x[j]+v[j]*dt;
      F[j]=-k[j]*x[j];
      v[j]=v[j]+F*dt/m[j];
      }
   }
```

About how many operations can a typical laptop computer do in a second?
How long, roughly, would you expect this code to run? Explain your answer!
If you wrote a python script for the same task, how long would you expect it to run?

[15.] In your next computational physics course, your instructor asks you to write a code to simulate the motion of the N air molecules in a small box. Of course you would need enough memory to store their positions and velocities. About how large N could you do using the memory on your laptop? How does that compare to Avogadro's number $N_A = 10^{23}$? Comment on the feasibility of simulating all the molecules in a macroscopic object.

PART THREE

[16.] What is the Taylor series for e^x?

[17.] Suppose you know the values of a function at three points $f(x), f(x-\Delta x), f(x+\Delta x)$, where Δx is small. How do you compute the second derivative of f at x?

[18.] You do a molecular dynamics simulation of a mass $m = 0.25$ kg on a spring $k = 100$ N/m. The initial position $x_0 = 0.7$ m and the initial velocity $v_0 = 20$ m/s. You want to check your code. What are some ways you can do it, without actually solving the differential equation for $x(t)$?
Note: Since you don't have a calculator to use, just put numbers in appropriate formulae but no need to evaluate.

[19.] Given a matrix $\mathcal{M}$. State *in words* what eigenvalues and eigenvectors of $\mathcal{M}$ are. No equations are allowed! I want to know the *physical meaning* of eigenvalues and eigenvectors.

PART FOUR
Do either [20] or [21] but not both!

[20.] Write a program (script) in python to solve the Kepler problem and make a plot of the orbital trajectory.

[21.] Write a program in C to solve the one dimensional diffusion equation.