13) (10 pts) Write the do loop for Newton's Method for root finding for the following equation. Do not write separate Fortran functions, just put the math in the do loop.

$$f(x) = x^2 + \cos(x) = 0$$
$$f'(x) = 2x - \sin(x) = 0$$

```
do
  if (abs (f(x)) .lt. Tolerance) Exit
  x = x + (f(x) /fPrime(x))
enddo
```

14) (10 pts) What is the **forward difference** approximation to the 1st derivative of the following function at $x = 3$ using a step size of $h = 0.5$ and a second answer using a step size $h = 0.1$? (No programming here, use the formula by hand just to get an answer.) **Circle** the one you think is most accurate to the actual derivative at 3. (Do the actual calculation using a calculator. Show your equations.)

$$y = x^2 + 2x - 13$$

$y' = 2x + 2$

$y' = 6 + 2 = 8$

$$f'(x) = \frac{f(x+h) - f(x)}{h}$$

$h = 0.5$

$y = 8.5$

$$h = 0.5 \rightarrow \frac{(3+\tfrac{1}{2})^2 + 2(3.5) \cancel{-13} - (9 + 6 \cancel{-13})}{0.5}$$

[✓ $h = 0.1$

$y = 8.1$]

$$h = 0.1 \rightarrow \frac{(3.1)^2 + 2(3.1) - (9 + 6)}{0.1}$$

15) (5 pts) Write a Fortran do loop that adds the **even** numbers from 2 to n. Just code the do loop. Assume n is already set.

```
do i = 2, n, 2
    even = even + i
enddo
```