Assignment_4

Due date: Friday, February 18 at 11:59pm in Gradescope

Before submission, ensure that the `.py` file is named:

`homework_4.py`

This assignment is worth 12 points.

In [1]:
```
import math
import numpy as np
import matplotlib.pyplot as plt

# Don't mind these lines just below; they're used to make the autograder wor
try:
    %matplotlib inline
except:
    pass
```

Problem 1 - (1.5 points) Write the solution to the following indefinite integrals using LaTeX. No credit will be awarded for work turned in any other format. Follow the model below.

$$\int \sin(x)\,dx = -\cos(x) + C$$

1.1:

$$\int \sin^2\left(\frac{5x}{2}\right) dx$$

1.2:

$$\int \left(\frac{1}{2x} - \frac{4}{x^3}\right) dx$$

1.3:

$$\int \frac{x}{\sqrt{x^2-4}}\,dx$$

Solution to Problem 1 : Write your answers in the lines below using LaTeX.

1.1:

$$\int \sin^2\left(\frac{5x}{2}\right) dx = \frac{1}{2}x - \frac{1}{10}\sin(5x) + C$$

1.2:

$$\int \left(\frac{1}{2x} - \frac{4}{x^3} \right) dx = \frac{1}{2}\ln(x) + \frac{2}{x^2} + C$$

1.3:

$$\int \frac{x}{\sqrt{x^2 - 4}}\, dx = (x^2 - 4)^{\frac{1}{2}} + C$$

Problem 2 - (4 points) Write Python functions that generate each of the following vectors; after writing the function, `print` the output to check your work and verify that the created vector matches your expectation.

2a. An array vector of 20 elements that range, equally spaced, from 0 to 2π; call the function `P2a`.

In [2]:
```
# Solution for Problem 2a

def P2a():
    vector_from_part_a = np.linspace(0, 2 * math.pi, 20)
    return vector_from_part_a
P2a()

    # Replace 'None' with what this function is supposed to return, same goe
```

Out[2]:
```
array([0.        , 0.33069396, 0.66138793, 0.99208189, 1.32277585,
       1.65346982, 1.98416378, 2.31485774, 2.64555171, 2.97624567,
       3.30693964, 3.6376336 , 3.96832756, 4.29902153, 4.62971549,
       4.96040945, 5.29110342, 5.62179738, 5.95249134, 6.28318531])
```

2b. An array vector that has the values -10 to 10 in steps of 0.5; call the function `P2b`.

In [3]:
```
# Solution for Problem 2b
def P2b():
    vector_from_part_b = np.arange(-10, 10.5, .5)
    return vector_from_part_b
P2b()
```

Out[3]:
```
array([-10. ,  -9.5,  -9. ,  -8.5,  -8. ,  -7.5,  -7. ,  -6.5,  -6. ,
       -5.5,  -5. ,  -4.5,  -4. ,  -3.5,  -3. ,  -2.5,  -2. ,  -1.5,
       -1. ,  -0.5,   0. ,   0.5,   1. ,   1.5,   2. ,   2.5,   3. ,
        3.5,   4. ,   4.5,   5. ,   5.5,   6. ,   6.5,   7. ,   7.5,
        8. ,   8.5,   9. ,   9.5,  10. ])
```

2c. An array vector that contains the elements with odd-numbered indices in the vector solution for part (2a); call the function `P2c`.

In [4]:
```
# Solution for Problem 2c
def P2c(vector_from_part_a):
    arrayIndex = []
    array = []
    for ii in range(1, vector_from_part_a.size, 2):
        arrayIndex.append(ii)
    array.append(vector_from_part_a[arrayIndex])
    return array
P2c(P2a())
```

Out[4]:
```
[array([0.33069396, 0.99208189, 1.65346982, 2.31485774, 2.97624567,
        3.6376336 , 4.29902153, 4.96040945, 5.62179738, 6.28318531])]
```

2d. An array vector that contains the elements with even-numbered indices in the vector solution for part (2b); call the function `P2d` .

In [5]:
```
# Solution for Problem 2d
def P2d(vector_from_part_b):
    arrayIndex = []
    array = []
    for ii in range(0, vector_from_part_b.size, 2):
        arrayIndex.append(ii)
    array.append(vector_from_part_b[arrayIndex])
    return array
P2d(P2b())
```

Out[5]:
```
[array([-10.,  -9.,  -8.,  -7.,  -6.,  -5.,  -4.,  -3.,  -2.,  -1.,   0.,
          1.,   2.,   3.,   4.,   5.,   6.,   7.,   8.,   9.,  10.])]
```

Problem 3 - (1.5 points) If the grading scale for a test is such that A=90-100, B = 80-89.9, C = 70-79.9, and any number below 70 is F. Write a function called `LetterGrade` that takes one scalar input representing the grade of a student. Your function should determine the letter grade of the student based on the input. For example, if you run your code with grade = 85.6, the function should return the following: `"You scored 85.6 on the test, your letter grade is B."`

Test your program with the following grades: 95.5, 34.9, 78.4, 79.9, 87.5 and present your output.

In [6]:
```python
# Solution for Problem 3
def LetterGrade(grade):
    if grade >= 90:
        letter = 'A'
    elif (grade <= 90) and (grade >= 80):
        letter = 'B'
    elif (grade <= 80) and (grade >= 70):
        letter = 'C'
    else:
        letter = 'F'
    string = 'You scored ' + str(grade) + ' on the test, your letter grade i
    return string

print(LetterGrade(95.5))
print(LetterGrade(34.9))
print(LetterGrade(78.4))
print(LetterGrade(79.9))
print(LetterGrade(87.5))
```

```
You scored 95.5 on the test, your letter grade is A.
You scored 34.9 on the test, your letter grade is F.
You scored 78.4 on the test, your letter grade is C.
You scored 79.9 on the test, your letter grade is C.
You scored 87.5 on the test, your letter grade is B.
```

Problem 4 - (3 points) Given two points $A(-4,-5)$ and $B(4,1)$ in XY coordinate system.

4a. Write a Python function called `distance` to calculate the distance between two points; test your code on the points A and B.

In [7]:
```python
# Solution for Problem 4a
A = [-4, -5]
B = [4, 1]
def distance(A, B):
    x1 = -4
    x2 = 4
    y1 = -5
    y2 = 1

    dist = math.sqrt ((B[0] - A[0]) ** 2 + (B[1] - A[1]) ** 2)
    #distance = math.sqrt ((x2 - x1) ** 2 + (y2 - y1) ** 2)
    return dist

distance(A, B)
```

Out[7]: 10.0

4b. Suppose that the line equation passing through points A and B is $y = mx + b$, where m is the slope and b is the y-intercept. Write a Python function called `slope_intercept` to find the slope and the y-intercept of the line; test your code on the points A and B, and make sure that the function returns the slope and y-intercept *in that order*.

In [8]:
```
# Solution for Problem 4b
A = [-4, -5]
B = [4, 1]
def slope_intercept(A, B):
    m = (B[1] - A[1]) / (B[0] - A[0])
    yInt = A[1] - (m * B[0] * -1)

    return m, yInt

print(slope_intercept(A, B))
```

```
(0.75, -2.0)
```

4c. Write a code to plot the graph of the line equation $y = mx + b$ from part (4b) on the interval $-4 \leq x \leq 4$. Provide appropiate labels for both axes (that means use `plt.xlabel` and `plt.ylabel` like you've seen in some of the handouts).

In [53]:
```
# Solution for Problem 4c
x = np.linspace(-4, 4, 100)
y = 0.75*x - 2
plt.plot(x, y, '-r', label='y = .75x - 2')
plt.title('Graph of y=2x+1')
plt.xlabel('x')
plt.ylabel('y')

plt.grid()
plt.show()
```

Problem 5 - (2 points) Write a Python code to produce each of the following plots.

For both parts (5a) and (5b), use enough points to make the graphs smooth and provide appropiate labels for all axes; you shouldn't be able to see any jagged lines on the curves. Further, label your axes like you did for (4c).

Hint : You can use the numpy package to produce the following:

- $\sin 6x$ $\longrightarrow$ `np.sin(6*x)`
- e^{-x} $\longrightarrow$ `np.exp(-x)`

5a. The graph for the function $x^2e^{-x^2}$ on the interval $[-5, 5]$ (remember that $-x^2$ is different from $(-x)^2$).

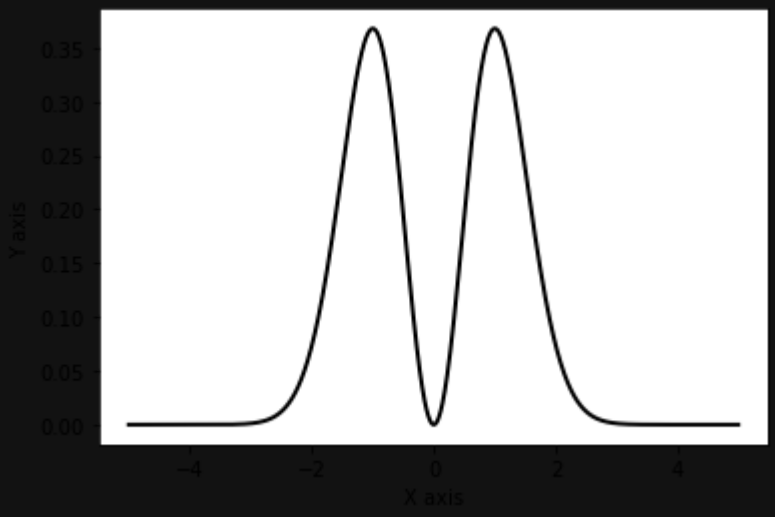
In [54]:
```
# Solution for Problem 5a
x = np.arange(-5, 5, .01)


plt.xlabel('X axis')
plt.ylabel('Y axis')

y1 = ((x * x) * (np.exp(-x**2)))

plt.plot(x, y1, linewidth = 2, color = 'black')

plt.show()
```



5b. The graphs for functions $\sin x$, $\sin 2x$, $\sin 3x$, $\sin 4x$, and $\sin 5x$ on the interval $[0, 2\pi]$ on one plot.

In [55]:

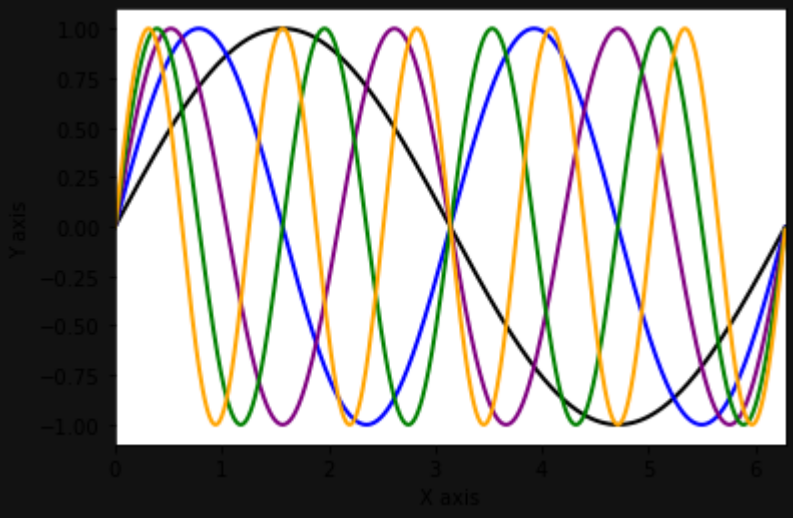
```
# Solution for Problem 5b
x = np.arange(0, 2 * np.pi, .01)

plt.xlim(0, 2 * np.pi)

plt.xlabel('X axis')
plt.ylabel('Y axis')

y1 = np.sin(x)
y2 = np.sin(2*x)
y3 = np.sin(3*x)
y4 = np.sin(4*x)
y5 = np.sin(5*x)

plt.plot(x, y1, linewidth = 2, color = 'black')
plt.plot(x, y2, linewidth = 2, color = 'blue')
plt.plot(x, y3, linewidth = 2, color = 'purple')
plt.plot(x, y4, linewidth = 2, color = 'green')
plt.plot(x, y5, linewidth = 2, color = 'orange')
plt.show()
```



In [ ]:

```
1
```