Problem 1.1

Vance Turnewitsch

Repeatedly roll two fair dice. Graph the cumulative distribution function $F(x)$ for the random variable X that sums their top face values on each roll.

Solution:

We are going to use pseudorandom number generation in Python for our die rolling and plotting.

In [37]:
```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

In [38]:
```
# This function "rolls" a single die
def rollDie():
    value = np.random.rand() # Unformly generates numbers [0,1)
    return int(value*6.0) # Scale from 1..6
```

In [39]:
```
# This function rolls two die and returns the sum of the result
def rollDice():
    die1 = rollDie()
    die2 = rollDie()
    return die1 + die2
```

In [40]:
```
# This is the cumulative distribution function that returns the
# probability that the measurement is less than x. Data should
# contain all the measurement results.
def F(x,data):
    size = 0.0
    count = 0.0
    for dat in data:
        if dat <= x: # Test if this datum is less than x
            count += 1.0
        size += 1.0
    return count/size
```

In [41]:
```
N = 1000 # Number of rolls
data = []
for i in xrange(N):
    data.append(rollDice()) # Roll our dice and record the result
```

In [42]:
```
x = np.linspace(2,12,11)
Fvect = []
for i in x:
    Fvect.append(F(i,data))
```

In [43]:
```
plt.plot(x,Fvect,"*")
plt.grid()
plt.xlabel("Die Face Sum")
plt.ylabel("F(Die Face Sum)")
plt.title("F(Die Face Sum) vs Die Face Sum")
```

Out[43]: <matplotlib.text.Text at 0x106b08290>