explosions.py

```
#!/usr/bin/env python3

import pygame, sys, time, random
from pygame.locals import *

ramp_one, ramp_two, ramp_three = None, None, None

wood_light = (166, 124, 54)
wood_dark = (76, 47, 0)
blue = (0, 100, 255)
dark_red = (166, 25, 50)
dark_green = (25, 100, 50)
dark_blue = (25, 50, 150)
black = (0, 0, 0)
white = (255, 255, 255)
```

explosions.py

```
width, height = 1024, 768
screen = None

maxRadius = 60
allExplosions = []
delay = 100   # number of milliseconds delay before generating a USEREVENT

class explosion:
    def __init__ (self, pos):
        self._radius = 1
        self._maxRadius = maxRadius
        self._increasing = True
        self._pos = pos
```

explosions.py

```
    def update (self):
        if self._increasing:
            pygame.draw.circle (screen, white, self._pos, self._radius, 0)
            self._radius += 1
            if self._radius == self._maxRadius:
                self._increasing = False
        else:
            pygame.draw.circle (screen, black, self._pos, self._radius, 0)
            self._radius -= 1
            if self._radius > 0:
                pygame.draw.circle (screen, white, self._pos, self._radius, 0)
            else:
                globalRemove (self)
```

explosions.py

```
def createExplosion (pos):
    global allExplosions
    allExplosions += [explosion (pos)]
    pygame.time.set_timer (USEREVENT+1, delay)

def globalRemove (e):
    global allExplosions
    allExplosions.remove (e)
```

explosions.py

```
def updateAll ():
    if allExplosions != []:
        for e in allExplosions:
            e.update ()
        pygame.display.flip ()
        pygame.time.set_timer (USEREVENT+1, delay)
```

explosions.py

-
  ```
  def wait_for_event ():
      global screen
      while True:
          event = pygame.event.wait ()
          if event.type == pygame.QUIT:
              sys.exit(0)
          if event.type == KEYDOWN and event.key == K_ESCAPE:
              sys.exit (0)
          if event.type == pygame.MOUSEBUTTONDOWN and event.button == 1:
              createExplosion (pygame.mouse.get_pos ())
          if event.type == USEREVENT+1:
              updateAll ()
  ```

explosions.py

```
def main ():
    global screen
    pygame.init ()
    screen = pygame.display.set_mode ([width, height])
    wait_for_event ()

main ()
```

Homework

- firstly get the `explosions.py` to work
- now comment each function
- comment each class and its use
- familiarise yourself with the game missile command
- see if you can extend this code to place six cities and 3 missile silos statically at the bottom of the screen

Homework 2

- write a Python program to print out all permutations of a string
- for example `ab`

-
  ```
  ab
  ba
  ```

Homework 2

- for example `abc`

-

```
abc
acd
bac
bca
cab
cba
```

- your program should be able to take any size string with unique characters and print out all permutations