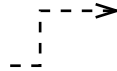


```
Square swapper 5000
+main(argc:integer,argv:string[])
```



**board**

```
-grid: cell array
-celg: lev class
-window: Xwindow type
-level: int
-inputfile: string
-movesremain: integer
-maxLevel: int
-score: int
+partscore: int
+highscore: int
-seed: int
-textMode: bool
-useFile: bool
-deletecell: int
-seq: string
-seqposition: int
-cleargrid(): void
-noLocked(): bool
-swapgrid(ab:cell,cd :cell): void
-drop(): void
-fillgrid(): void
-hasmatch(r:int,c:int): bool
-Leftmatches(row:int,col:int): int
-Rightmatches(row:int,col:int): int
-Upmatches(row:int,col:int): int
-Downmatches(row:int,col:int): int
-psychedelic(row:int,col:int): bool
-upright(row:int,col:int): bool
-lateral(row:int,col:int): bool
-basic(row:int,col:int): bool
-unstable(row:int,col:int): bool
-matchexist(): bool
-exp(num:int,chain:int): int
-deletegrid(x:int,y:int,z:int,num:int): void
-getpoints(squares:int,chain:int): int
-action(): void
-draw(): void
-isvalidswap(r:int,c:int,direction:int): bool
-isamatch(color:char,r:int,c:int,direction:int): bool
-InBound(x:int,y:int): bool
+createlevel0(): void
+createlevel1(): void
+createlevel2(): void
+board(textMode:bool,seed:int,inputfile:string,useFile:bool,startlevel:int,): ctor
+changeFile(newFile:string): void
+setwindow(window:Xwindow): void
+-board(): Dtor
+init(): void
+swap(x:int,y:int,direction:int): void
+hint(): string
+scramble(): void
+levelup(): void
+leveldown(): void
+restart(): void
```

**lev**

```
-level: int
-seed: int
+ab: cell type
+levelonesequene: string
+levelonescounter: int
+lev(seed:int): ctor
+-lev(): Dtor
+updatelevel(level:int): void
+generate(ab:cell type): void
```

**Xwindow**

```
+Xwindow(wide:integer,height:integer): ctor
+-window(): dtor
+fillRectangle(x:int,y:int,wide:int,height:int,colour:int): void
+drawString(x:int,y:int,msg:string,colour:int): void
+drawBigString(x:int,y:int,msg:string,colour:int): void
```

**cell**

```
-state: char
+square: char
+colour: char
+col: int
+row: int
+
+cell(): ctor
+-cell(): dtor
+setcell(state:char,square:char,colour:char): void
+seccoor(c:int,r:int): void
+getstate(): char
+getsquare(): char
+getcolour(): char
+draw(window:Xwindow): void
```