```
/*
 * Provisional PGDir and page tables setup
 *
 * for mapping two linear address ranges to the same physical address range
 *
 *  + Linear address ranges:
 *              -   User mode: $i\times{}4M\sim{}(i+1)\times{}4M-1$
 *              - Kernel mode: $3G+i\times{}4M\sim{}3G+(i+1)\times{}4M-1$
 *  + Physical address range: $i\times{}4M\sim{}(i+1)\times{}4M-1$
 */
typedef unsigned int PTE;
PTE *pg = pg0;      /* physical address of pg0 */
PTE pte = 0x007;    /* 0x007 = PRESENT+RW+USER */
for(i=0;;i++){
  swapper_pg_dir[i] = pg + 0x007;          /* store identity PDE entry */
  swapper_pg_dir[i+page_pde_offset] = pg + 0x007; /* kernel PDE entry */
  for(j=0;j<1024;j++){                    /* populating one page table */
    pg[i*1024 + j] = pte;                 /* fill up one page table entry */
    pte += 0x1000;                        /* next 4k */
  }
  if(pte >= ((char*)pg + i*1024 + j)*4 + 0x007 + INIT_MAP_BEYOND_END)
    {
      init_pg_tables_end = pg + i*0x1000 + j;
      break;
    }
  }
```