Modified Player Initialization Function (Changes in Blue)

```
int32_t player_init
    (struct player_t* p,
     const char* n,
     const char* pswd, int32_t p_age)
{
    int32_t i;
    for (i = 0; 31 > i && '\0' != n[i];
         i++) {
        p->name[i] = n[i];
    }
    p->name[i] = '\0';
```


Finish Initializing Fields Based on Parameters

```
for (i = 0;
     19 > i && '\0' != pswd[i];
     i++) {
    p->password[i] = pswd[i];
}
p->name[i] = '\0';
p->age = p_age;
```

Copy up to 19 characters.

Terminate with NUL.

Copy age into player struct.

Initialize Remaining Fields with Constant Values

```
    p->num_games = 0;
    for (i = 0; 16 > i; i++) {
        p->score_dist[i] = 0;
    }
    p->game = NULL;
    return 1;
}
```

no games played yet

Return success.

no scores yet

no game being played

Let's Write Two More Functions for Players

Let's also write

- **`player_new_game`**, for when a player starts a game, **and**
- **`player_finish_game`**, for when a player finishes a game.

Both will **take a `struct player_t*`** as one **parameter**.

Both will **return an `int32_t`**: 0 for failure, 1 for success.

