

```
int proc1() {
    ...
    return ...;
}
```

```
void main(){
    ...
    int n = proc1();
    printf("%d\n", n);
    ...
}
```

thread-1

main

proc1

main

W tym czasie instrukcje należące do main się nie wykonują. Inaczej: main czeka na powrót z wywołania proc1.

Definicja synchroniczności w programowaniu

```
async function proc1() {
    ...
    return 55;
}

async function main() {
    ...
    proc1();
    ...
}
```

Wywołaniu asynchronicznemu towarzyszy powstanie specjalnego obiektu Future, który jest reprezentantem wartości wyprodukowanej przez proc1 w bliżej nieokreślonej przyszłości.

main

proc1

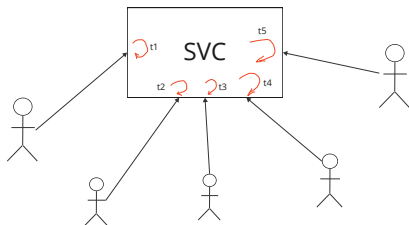
main

55

55

Procedura main nie czeka na powrót z wywołania proc1.
Obydwie procedury muszą więc działać równoległe do siebie, a więc w osobnych wątkach.

Definicja asynchroniczności w programowaniu



Blocking IO