foo.h

```
class foo {
  public:
    int do();
};
```

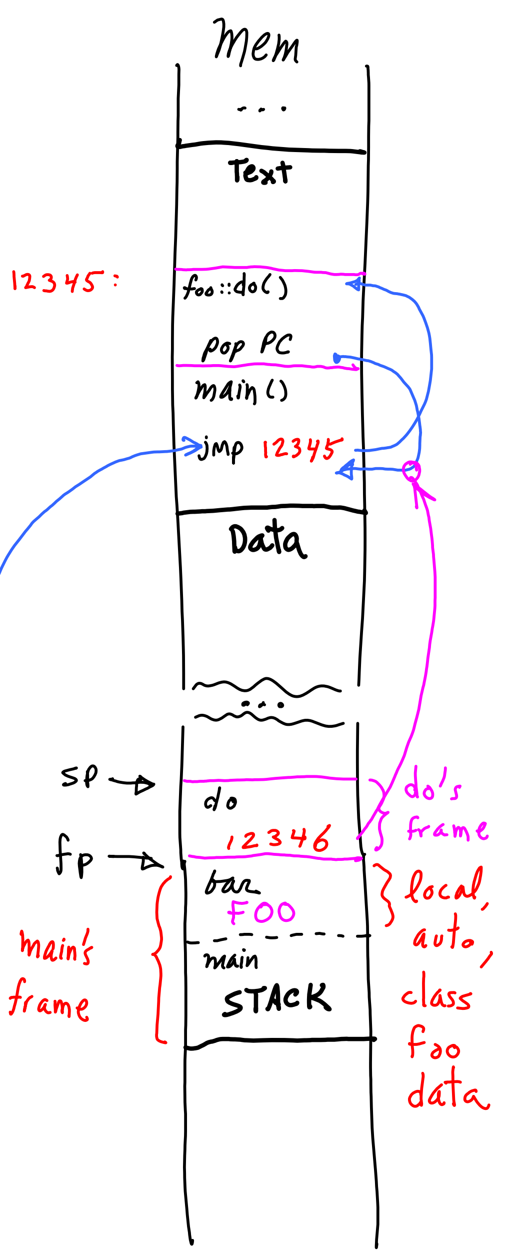
foo.cpp

```
int foo::do() {
    return(0);
}
```

App.c

```
#include "foo.h"
int main() {
    foo bar;
    bar.do();
    return (0);
}
```

```
gcc -o foo.o foo.cpp    } create foo.o,
                           => library  Foo.a
gcc -o App App.c foo.o  }   -l Foo
```



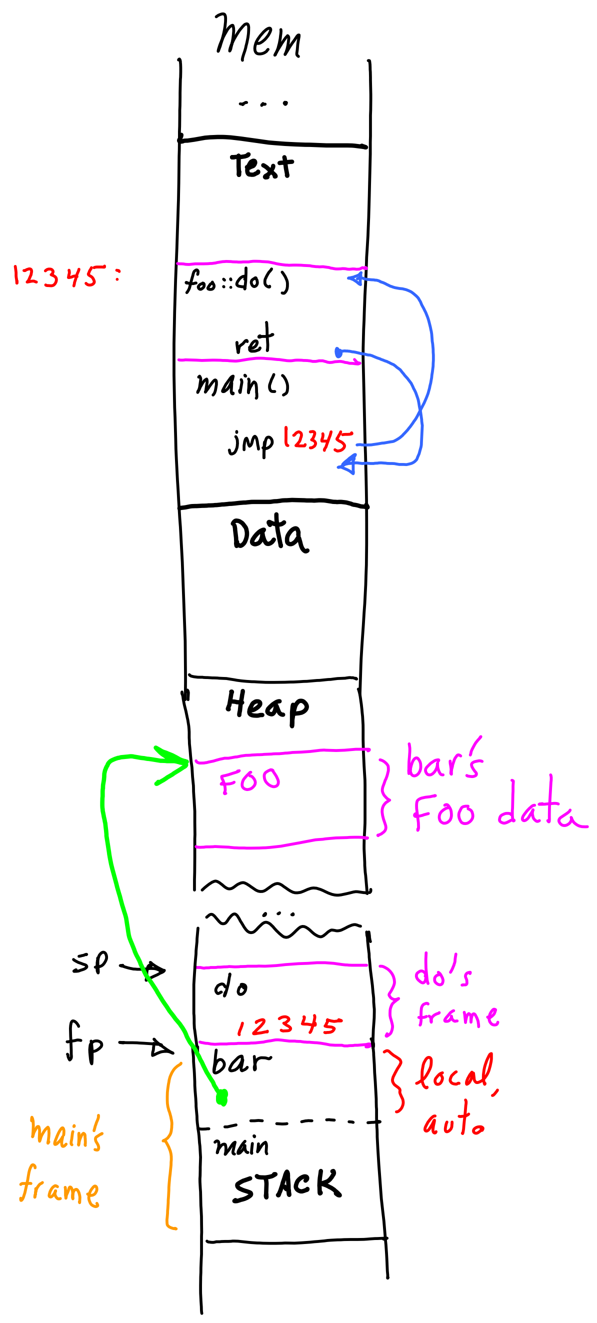
compile time: known address of function bar.do() ==> jmp 12345

run time: unknown return address is on top of stack ==> pop PC (= 12346)

```
(gdb) sim
    ) l
    ) b main
    ) r
    ) l
    ) b Event.insertEvent
    ) c
    > bt
    ) n      (don't 's' into new())
```

```
) p root  (in context of EventQ
) n ...
) p queue.root  (in main's context)
) n ...   (hit bp again)
) info b  ( see bps )

        edit sim.cpp
          fix order

) n ... till crashes
```

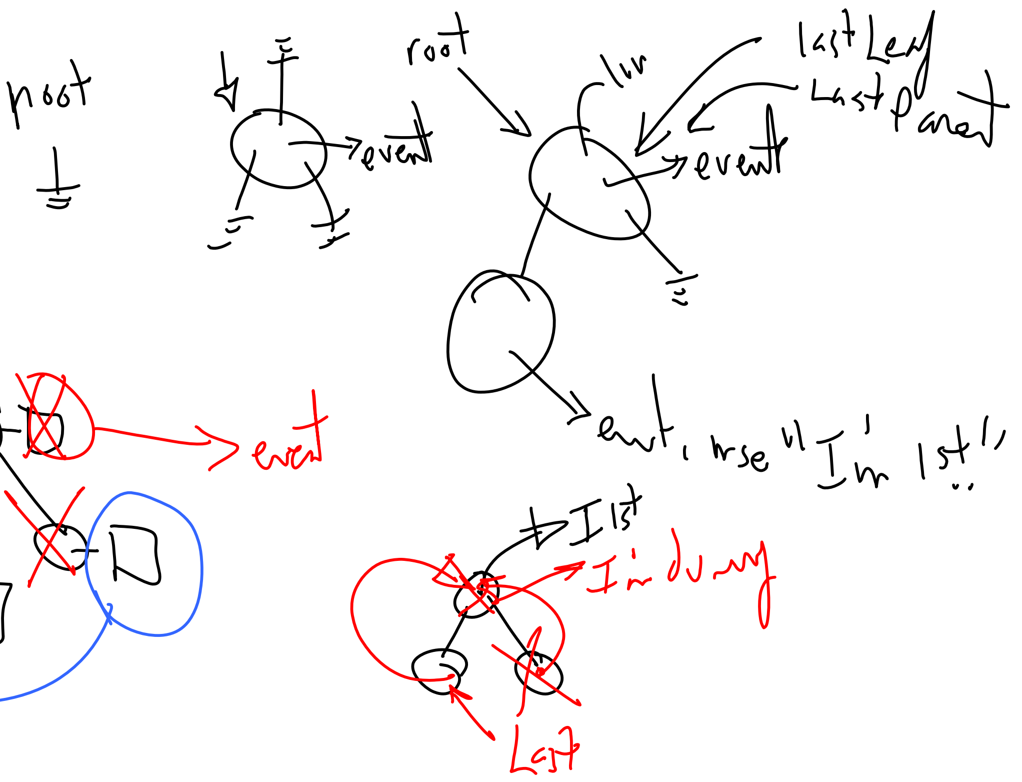
when did lastLeaf get updated last?

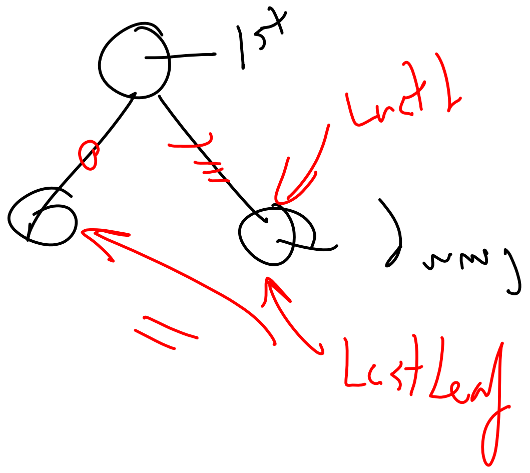


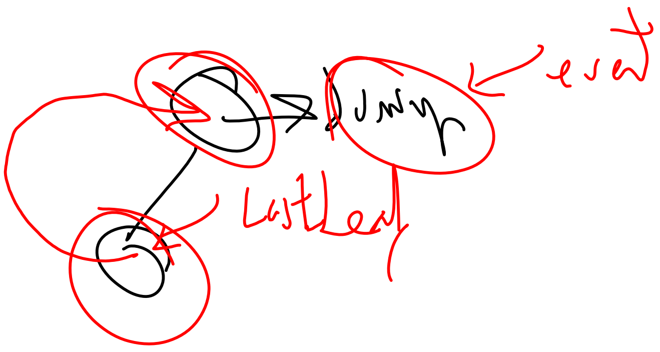
```
#include "foo.h"
int main() {
    foo *bar = new foo;
    *bar.do();
    return (0);
}
```

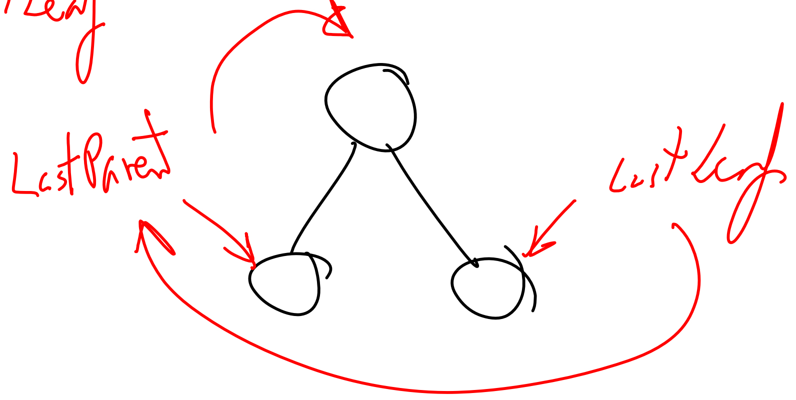
```
class foobar: foo {
            public doIt();
}
```

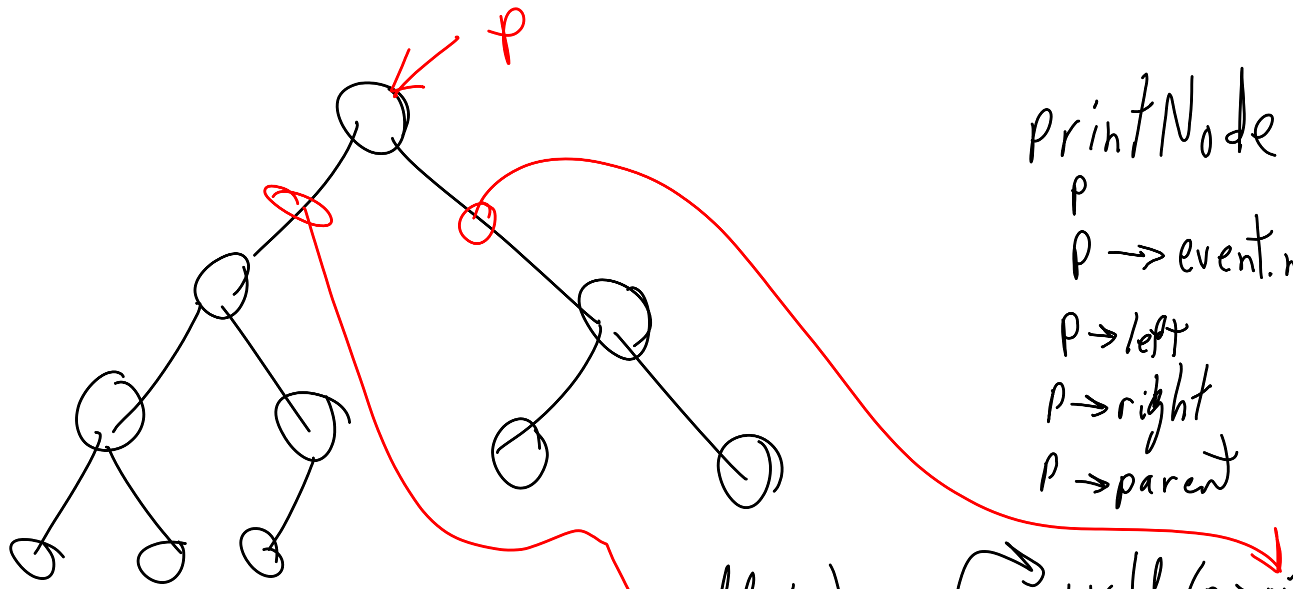
```
class Thing {
        virtual void show() const;
                void hello() const;
}


class Bunny : Thing {                    class fox : Thing {

        void show() const,                       void show() const;

};                                       };

void Bunny::show( ) const {              void Fox::show() const {

}                                        }


    Thing *ptr;

     ptr = new Bunny;
     list.insert(ptr);
    ptr = new Fox;
    list.insert(ptr);

   ptr = list.getNext();
   ptr -> show();
   ptr -> hello();


   ptr = list.getNext();
   ptr -> show();
   ptr -> hello();
```


root
root
cur
lastLeaf
LastParent
event
event
event
event, horse "I'm 1st"
I 1st
I'm dummy
Last









```
printNode(p)
  p
  p->event.msg
  p->left
  p->right
  p->parent
```

```
walk( p )
  if (p==NULL) return;

walk(p)
printNode(p)
walk(p->left)
walk(p->right)
return;
```

does it work for all Tree size $\leq n$?

if it work size $\leq n \Rightarrow$ work size $n+1$

(work size $= 0$)

$$\frac{(w == 0) \qquad (w == 0) \Rightarrow (w == 1)}{(w == 1)}$$

$$\frac{(w == 1) \qquad (w == 0) \land (w == 1) \Rightarrow (w == 2)}{(w == 2)}$$