30. Combat Function Category

戦闘に関するファンクション

Functions

ActionCounterSpell

呼出者の戦闘状態を「呪文相殺」モードにします。

ActionUseFeat

oTargetのフィート（特技）を使用させます。

AssociateCheck

objectがヘンチマン（PCの傭兵）かそうでないかを調べます。

BashDoorCheck

ヘンチマンが扉を殴り続けさせる為に DetermineCombatRoundの中で使われます。

CheckEnemyGroupingOnTarget

ターゲット周辺の球体状の範囲内の敵の数を取得します。

CheckFriendlyFireOnTarget

ターゲット周辺の球体状の範囲内の友好的クリーチャーの数を取得します。

chooseTactics

職業に特有の戦術を決定します。

DetermineClassToUse

NPCのキャラクタークラスを見て、戦闘中にどれを使用すべきであるか決めます。

DetermineCombatRound

特定のラウンドの間にとるNPCの行動を決定します。

EffectDamageReduction

ダメージ軽減effectを作成します。

EffectDamageResistance

目標をあるタイプのダメージにもっと抵抗するeffectを返します。

EffectDamageShield

攻撃が成功したものにダメージを与えるダメージシールド（エレメンタルシールド）を目標の回りに作成します。

EffectDeath

デスeffectを作成します。

GetGoingToBeAttackedBy

現在の戦闘ラウンドで他のクリーチャーを攻撃しているクリーチャーを取得します。

GetIsFighting

クリーチャーが戦闘中かどうかを得ます。

GetLastHostileActor

objectに対し最後に敵対行為を実行した、プロットクリーチャーではない生存者を得ます。

JumpToWeakestEnemy

最もダメージを受けている敵にジャンプする

SearchForPlayer

一番近いPCを探し攻撃する。

TalentRangedAttackers

射程ある攻撃法で攻撃されている間の、呪文詠唱者であるNPCの行動方針を決定します。

See Also

Function Categories

ActionCounterSpell(object)

呼出者の戦闘状態を「呪文相殺」モードにします。

```
void ActionCounterSpell(  
    object oCounterSpellTarget  
);
```

Parameters

oCounterSpellTarget
「呪文相殺」で対抗する相手Object

Description

呼出クリーチャーを「呪文相殺」型戦闘モードにさせ、oCounterSpellTargetが呪文を唱えるまで待機します。

Remarks

呼出クリーチャーがこの関数を呼び出した後、別の行動をとった場合、「呪文相殺」モードは取り消されます。

呼出クリーチャーが「呪文相殺」モードに入り、oCounterSpellTargetを「呪文相殺」を試みる前には、ある基準が満たされてなければなりません。：

「呪文相殺」モードが使えるようになる前に、呪文を唱えることのできる職業で、唱えられるレベルになってなければなりません。

oCounterSpellTargetが唱えようとしている呪文が識別できる、十分な 呪文学 技能を持ってなければなりません。

「呪文相殺」で消費される適用可能な呪文を持っていなければなりません。（NWNマニュアルの109ページを見てください。（日本語版では134ページ））

上記の条件がすべて満たされても、「呪文相殺」の成功は保証されていません。

さらに「呪文相殺」モードが準備行動と見なされることに注意してください。それは単に相手のラウンドが始まる前に、効力を得るということを意味します。相手が呪文を唱えている間に「呪文相殺」を始めても、「呪文相殺」を試みることはできません。

[未知：「呪文相殺」モードにするクリーチャー指揮棒の作り方を知っていますか？

DetermineCombatRound()は動作しません。プレイヤーが操作するクリーチャーでそれを行えば、「呪文相殺」モードは無期限に続くでしょう。]「呪文相殺」モードは1ラウンド（6秒間）続きます。

Version

1.22

See Also

functions: ActionCastSpellAtLocation | ActionCastSpellAtObject

categories: Action on Object Functions | Combat Functions | Combat Actions Functions

author: Troels Therkelsen, editor: Jeff Lindsey, JP team: Rainie

ActionUseFeat(int, object)

*oTarget*のフィート（特技）を使用させます。

```
action ActionUseFeat(  
    int nFeat,  
    object oTarget  
);
```

Parameters

nFeat
FEAT_*

oTarget
特技の対象になるobject

Description

ActionUseFeatは、クリーチャーに指定した特技を目標のobjectに対して使用させます（適切な場合）。クリーチャーがその特技を持っていない場合、この関数はなにもしません。

Remarks

この行動はPCにさせることはできません。

Version

1.22

Example

```
//NPCに「レイ・オン・ハンズ（Lay on Hands）」を  
//会話中にさせるスクリプト：  
ActionUseFeat(FEAT_LAY_ON_HANDS, GetPCSpeaker());
```

See Also

functions: ActionUseSkill | ActionUseTalentAtLocation | ActionUseTalentOnObject | GetHasFeatEffect
categories: Action on Object Functions | Combat Functions | Talents/Skills/Feats Functions
constants: FEAT_* Constants

author: Iskander Merriman, JP team: Rainie

AssociateCheck(object)

objectがヘンチマン（PCの傭兵）かそうでないかを調べます。

```
int AssociateCheck(  
    object oCheck  
);
```

Parameters

oCheck
ヘンチマンかどうかの状態を調べたいobject

Description

これはエリア内の友好的なクリーチャーの数決定の一部として使用される一般的なAIスクリプトの関数です。

Remarks

下記に示すプログラムのように、AssociateCheckはオブジェクト比較を備えたGetAssociate()を含みます。

Requirements

```
#include "nw_i0_generic"
```

Version

1.22

Example

```
// これはnw_i0_generic内のAssociateCheckプログラムである  
int AssociateCheck(object oCheck)  
{  
    object oHench = GetAssociate (ASSOCIATE_TYPE_HENCHMAN);  
    if(oCheck != oHench)  
    {  
        return TRUE;  
    }  
    return FALSE;  
}
```

See Also

functions: CheckFriendlyFireOnTarget | GetAssociate
categories: Combat Functions | Core AI Functions

author: Iskander Merriman, editor: Dave Robinson, JP team: Rainie

BashDoorCheck(object)

ヘンチマンが扉を殴り続けさせる為にDetermineCombatRoundの中で使われます。

```
int BashDoorCheck(  
    object oIntruder = OBJECT_INVALID  
);
```

Parameters

oIntruder
(デフォルト: OBJECT_INVALID)

Description

もしそのエリアに敵対者がいなければ、ヘンチマンがドアを殴り続けさせておくために、このチェックはDetermineCombatRound()の中でなされます。

Remarks

もし独自のカスタムAIのためにDetermineCombatRoundを書き直していなければ、恐らくこの関数を必要としません。

Requirements

```
#include "nw_i0_generic"
```

Version

1.22

Example

```
// nw_i0_genericのヘッダより:  
/*  
*****  
WARNING THIS SCRIPT IS CHANGED AT YOUR PERIL  
*****  
  
This is the master generic script and currently  
handles all combat and some plot behavior  
within NWN. If this script is tampered  
with there is a chance of introducing game  
breaking bugs. But other than that enjoy.  
*/  
// If you're confident that using or modifying BashDoorCheck  
// is the right thing to do, you don't need sample code.
```

See Also

functions: DetermineCombatRound
categories: Combat Functions | Core AI Functions

author: Iskander Merriman, editor: Tom Cassiotis, JP team: Rainie

CheckEnemyGroupingOnTarget(object, float)

ターゲット周辺の球体状の範囲内の敵の数を取得します。

```
int CheckEnemyGroupingOnTarget(  
    object oTarget,  
    float fDistance = 5.0  
);
```

Parameters

oTarget

敵数をチェックする球体範囲の中心になる対象のobject

fDistance

敵数をチェックする球体状の半径（デフォルト：5.0）

Description

oTargetを中心とした半径fDistanceを持つ球体範囲内の敵対ファクションの総数を返します。これは CheckFriendlyFireOnTargetの反対です。

Remarks

この関数はnw_i0_generic.nssの3463行目にあります。

NPCのAIには非常に使える関数です。1つのクリーチャー周辺の全てのターゲットを1周循環し、呼出objectが敵対と見なす者の数を得ます。

Requirements

```
#include "nw_i0_generic"
```

Version

1.22

Example

```
// NPCに1以上の敵の集団を見渡させ、友好的なクリーチャーが  
// 範囲内にないことを確かめます。次に、ファイアボール呪文を  
// 持っている場合、それを発射させる  
void main()  
{  
    oTarget = GetNearestCreature(CREATURE_TYPE_REPUTATION, REPUTATION_TYPE_ENEMY,  
    OBJECT_SELF, 1, CREATURE_TYPE_PERCEPTION, PERCEPTION_SEEN);  
    if(GetIsObjectValid(oTarget))  
    {  
        int nEnemy = CheckEnemyGroupingOnTarget(oTarget);  
        int nFriendlies = CheckFriendlyFireOnTarget(oTarget);  
        if( nFriendlies < 0 && nEnemy > 1)  
        {  
            ActionCastSpellAtObject(SPELL_FIREBALL,oTarget);  
        }  
    }  
}
```

See Also

functions: CheckFriendlyFireOnTarget

categories: Combat Functions | Core AI Functions

author: Iskander Merriman, editor: John Shuell, JP team: Rainie

CheckFriendlyFireOnTarget(object, float)

ターゲット周辺の球体状の範囲内の友好的クリーチャーの数を取得します。

```
int CheckFriendlyFireOnTarget(  
    object oTarget,  
    float fDistance = 5.0  
);
```

Parameters

oTarget

友好者数をチェックする球体範囲の中心になる対象のobject

fDistance

友好者数をチェックする球体状の半径 (デフォルト: 5.0)

Description

oTargetを中心とした半径fDistanceを持つ球体範囲内の友好的ファクションの総数を返します。

Remarks

呼出者のヒットダイスの半分に満たない友好的クリーチャーは数えられません。

コアAI内で、保護系や増強系または有害な範囲呪文を使うかどうか決定するために使われています。目標範囲内に友好者がいる間は、ヘンチマンが無差別に有害な範囲呪文を使うことを防ぎます。

Requirements

```
#include "nw_i0_generic"
```

Version

1.26

Example

```
// NPCに1以上の敵の集団を見渡させ、友好的なクリーチャーが  
// 範囲内にないことを確かめます。次に、ファイアボール呪文を  
// 持っている場合、それを発射させる  
void main()  
{  
    oTarget = GetNearestCreature(CREATURE_TYPE_REPUTATION, REPUTATION_TYPE_ENEMY,  
    OBJECT_SELF, 1, CREATURE_TYPE_PERCEPTION, PERCEPTION_SEEN);  
    if(GetIsObjectValid(oTarget))  
    {  
        int nEnemy = CheckEnemyGroupingOnTarget(oTarget);  
        int nFriendlies = CheckFriendlyFireOnTarget(oTarget);  
        if( nFriendlies <= 0 && nEnemy >= 1)  
        {  
            ActionCastSpellAtObject(SPELL_FIREBALL,oTarget);  
        }  
    }  
}
```

See Also

functions: AssociateCheck | CheckEnemyGroupingOnTarget | GetAttackTarget | GetAttemptedAttackTarget | GetIsFriend | GetIsReactionTypeFriendly
categories: Combat Functions | Core AI Functions

author: Iskander Merriman, editor: Charles Feduke, JP team: Rainie

chooseTactics(object)

職業に特有の戦術を決定します。

```
int chooseTactics(  
    object oIntruder  
);
```

Parameters

oIntruder

戦術で対抗するクリーチャー

Description

以前はDetermineCombatRoundの一部でしたが、この機能性は目的を分かりやすくするためにDetermineCombatRoundから派生しました。

Requirements

```
#include "nw_i0_generic"
```

Version

1.30

See Also

functions: DetermineCombatRound

categories: Combat Functions | Core AI Functions

author: Charles Feduke, JP team: Rainie

DetermineClassToUse()

NPCのキャラクタークラスを見て、戦闘中にどれを使用すべきであるか決めます。

```
int DetermineClassToUse();
```

Description

DetermineCombatRound関数の中で使われたNPCのクラスを返します。
3つのクラススロットをすべて調べて、3つの中でどれが一番高いかを比較して決定します。

Remarks

どのクラスを送るかを任意に選択します。

Requirements

```
#include "NW_O2_CONINCLUDE"
```

Version

1.22

See Also

functions: DetermineCombatRound

categories: Combat Functions | Get Data from Creature Functions

author: Jody Fletcher, JP team: marshall

DetermineCombatRound(object, int)

特定のラウンドの間にとるNPCの行動を決定します。

```
void DetermineCombatRound(  
    object oIntruder = OBJECT_INVALID,  
    int nAIDifficulty = 10  
);
```

Parameters

oIntruder
(デフォルト: OBJECT_INVALID)

nAIDifficulty
使用しません (デフォルト : 10)

Description

この関数は一般的に含まれる主要スクリプトから呼出される主要な関数です。この関数は実際にスクリプトを書く代わりに使われます。この関数は、NPCの呪文詠唱能力やその他の可能なアクションを検査します。

Requirements

```
#include "NW_I0_GENERIC"
```

Version

1.28

See Also

functions: [ActionAttack](#) | [chooseTactics](#) | [DetermineAssociateCombatRound](#) | [DetermineClassToUse](#)

categories: [Combat Functions](#) | [Core AI Functions](#)

events: [OnPhysicalAttacked Event](#)

author: Jody Fletcher, editor: Charles Feduke, additional contributor(s): Tim Stokman, JP team: Rainie

EffectDamageReduction(int, int, int)

ダメージ軽減effectを作成します。

```
effect EffectDamageReduction(  
    int nAmount,  
    int nDamagePower,  
    int nLimit = 0  
);
```

Parameters

nAmount

軽減されるダメージの量

nDamagePower

DAMAGE_POWER_*グループで定義された軽減されるダメージのタイプ

nLimit

effectが消滅する前に吸収できるダメージの最大量。0にすると無限大。(Default: 0)

Description

このeffectを適用すると目標が受けるダメージの量が下がります。このeffectは指定されたまたは無限にダメージ量を散らせます。

Remarks

このeffectはApplyEffectToObject()の中で使用することで効果を表します。使用する場合は、そちらもご覧下さい。

Version

1.22

See Also

functions: EffectDamageResistance

categories: Combat Functions | Effects Functions

constants: DAMAGE_POWER_* Constants

author: John Shuell, JP team: akito

EffectDamageResistance(int, int, int)

目標のあるタイプのダメージにもっと抵抗するeffectを返します。

```
effect EffectDamageResistance(  
    int nDamageType,  
    int nAmount,  
    int nLimit = 0  
);
```

Parameters

nDamageType

DAMAGE_TYPE_*グループで定義した抵抗するダメージのタイプ

nAmount

目標がダメージを受けるごとに吸収するダメージの量

nLimit

このeffectが消える前に吸収できるダメージの最大量。0にすると無限に吸収します。(Default: 0)

Description

nLimitの上限（nLimitが0なら無限）まで、nDamageTypeタイプのnAmount量を取り去るダメージ抵抗effectを返します。

Remarks

このeffectはApplyEffectToObject()の中で使用することで効果を表します。使用する場合は、そちらもご覧下さい。

Version

1.22

See Also

functions: EffectDamageDecrease | EffectDamageReduction | EffectDamageShield

categories: Combat Functions | Effects Functions

constants: DAMAGE_TYPE_* Constants

author: John Shuell, JP team: akito

EffectDamageShield(int, int, int)

攻撃が成功したものにダメージを与えるダメージシールド（エレメンタルシールド）を目標の回りに作成します。

```
effect EffectDamageShield(  
    int nDamageAmount,  
    int nRandomAmount,  
    int nDamageType  
);
```

Parameters

nDamageAmount

相手によって攻撃が成功した後に、目標を攻撃した者に適用されるダメージの量。

nRandomAmount

DAMAGE_BONUS_*定数グループで定義されたシールドによって与えられる基本ダメージに加えられる量。

nDamageType

DAMAGE_TYPE_*定数グループで定義された攻撃者に与えられるダメージのタイプ。

Description

ダメージシールド（エレメンタルシールド）effectを目標に適用すると、nDamageTypeで指定したタイプのダメージ(nDamageAmount + nRandomAmount)を、目標と混戦している攻撃が成功した者に返す。

Remarks

このeffectはApplyEffectToObject()の中で使用することで効果を表します。使用する場合は、そちらもご覧ください。

Version

1.22

See Also

functions: EffectDamageResistance

categories: Combat Functions | Effects Functions

constants: DAMAGE_BONUS_* Constants | DAMAGE_TYPE_* Constants

author: John Shuell, JP team: akito

EffectDeath(int, int)

デスeffectを作成します。

```
effect EffectDeath(  
    int nSpectacularDeath = FALSE,  
    int nDisplayFeedback = TRUE  
);
```

Parameters

nSpectacularDeath

もしこれがTRUEなら、このeffectを適用したクリーチャーは特異な死に方をします。(Default: FALSE)

nDisplayFeedback

(Default: TRUE)

Description

この新しいeffectが目標に適用されると、それらのヒットポイントは0に設定され、death animationを行い(もしnSpectacularDeathがTRUEなら、NPCの死体は残るようになります。)、そして、そのクリーチャーは命令を受け付けなくなります。--それらは動くことも、攻撃することも、魔法を唱えることも、もう何もできません。

ドアや配置オブジェクトに適用すると、それらは破壊され、shatter animationを行います。

death effectは'死ぬ'ように目標を設定し、それから自身を削除する瞬間のeffectです。(ヒール、リザレクションなどの効果とともに、少数の不可逆なeffectのうちの1つです。)

オブジェクトは死に(破壊され)、永久に消滅します。

クリーチャーを生き返らせたいなら、EffectResurrection()を使う必要があります。

Remarks

このeffectはApplyEffectToObject()の中で使用することで効果を表します。使用する場合は、そちらもご覧下さい。

Version

1.22

See Also

categories: [Combat Functions](#) | [Effects Functions](#)

author: John Shuell, JP team: akito

GetGoingToBeAttackedBy(object)

現在の戦闘ラウンドで他のクリーチャーを攻撃しているクリーチャーを取得します。

```
object GetGoingToBeAttackedBy(  
    object oTarget  
);
```

Parameters

oTarget
攻撃されているクリーチャー

Description

*oTarget*を攻撃しているクリーチャーを返します。*oTarget*が有効なクリーチャーではない、または*oTarget*が攻撃されていない場合は、OBJECT_INVALIDを返します。

Remarks

全ての戦闘ラウンドの終了時に、この値は消去されます。（このクリーチャーがヘンチマンである時）主人であるクリーチャーから「攻撃しろ」という命令を得た時を除いては、どんな場合でも使用すべきではありません。

Version

1.22

See Also

categories: [Combat Functions](#) | [Core AI Functions](#) | [Henchmen/Familiars/Summoned Functions](#)

author: Tom Cassiotis, JP team: Rainie

GetIsFighting(object)

クリーチャーが戦闘中かどうかを得ます。

```
int GetIsFighting(  
    object oFighting  
);
```

Parameters

oFighting
使用しません。

Description

この関数をスクリプトされたNPCが、他の対象に呪文をかけたり、攻撃することを試みた場合、TRUEを返し、そうでなければFALSEを返します。

Remarks

クリーチャーが戦っているかどうか決めるために、GetAttemptedAttackTarget()およびGetAttemptedSpellTarget()を使用します。

Requirements

```
#include "NW_I0_GENERIC"
```

Version

1.22

Example

```
// この関数用のソースプログラムです  
  
int GetIsFighting(object oFighting)  
{  
    object oAttack = GetAttemptedAttackTarget();  
    object oSpellTarget = GetAttemptedSpellTarget();  
  
    if(GetIsObjectValid(oAttack) || GetIsObjectValid(oSpellTarget))  
    {  
        return TRUE;  
    }  
    return FALSE;  
}
```

See Also

functions: [GetAttackTarget](#) | [GetAttemptedAttackTarget](#)
categories: [Combat Functions](#) | [Core AI Functions](#)

author: Tom Cassiotis, editor: John Shuell, JP team: Rainie

GetLastHostileActor(object)

objectに対し最後に敵対行為を実行した、プロットクリーチャーではない生存者を得ます。

```
object GetLastHostileActor(  
    object oVictim = OBJECT_SELF  
);
```

Parameters

oVictim

返したい現在敵であるobject (デフォルト : OBJECT_SELF)

Description

GetLastAttacker()、GetLastDamager()、GetLastSpellCaster() (敵対呪文時) または GetLastDisturbed() (クリーチャーがすりした場合) と同様の呼ばれた最後のObjectを返します。注意 : 返り値は、単に常に次のとおりかもしれません:*。プロットキャラクターであるクリーチャー*はこの値に設定*をたないでしょう。AoE objectが、それらがこの値として登録されている場合、AoEの呼出者を返すでしょう。そうでなければINVALID_OBJECT_ID*を返します。トラップはtrap.*であるクリーチャーを返しません。この値は他のクリーチャーでないobject.*に上書きされることはありません。この値は死亡/破壊されたクリーチャーを返すことはありません。

Remarks

非常に使える関数です。明確な行動の種類を限定せずに敵対行為を行ったobjectを知りたいと思うことが時々あるだろう。様々な攻撃者を得る関数を一つにひっくるめた関数です。クリーチャーではない物を返さないの、殆どは戦闘目標を決定するために使用されます。(言い換えれば、罠を仕掛けた箱を表すobjectや罠等は返しません。) また、プロット設定のクリーチャーの場合も返しません。それはこの関数が、元はプロットNPCが他のNPCによって狙われないようにするAIの中で排他的に使用されていたことからきています。

Version

1.22

See Also

functions: [GetLastAttacker](#) | [GetLastDamager](#) | [GetLastDisturbed](#) | [GetLastKiller](#) | [GetLastSpellCaster](#) | [GetLastWeaponUsed](#) | [GetTotalDamageDealt](#) | [RespondToShout](#)
categories: [Combat Functions](#) | [Core AI Functions](#) | [Get Data from Object Functions](#) | [Henchmen/Familiars/Summoned Functions](#) | [Targeting Functions](#)

author: John Shuell, JP team: Rainie

JumpToWeakestEnemy(object)

最もダメージを受けている敵にジャンプする

```
void JumpToWeakestEnemy(  
    object oTarget  
);
```

Parameters

oTarget

敵を見つけるファクションのメンバー

Description

最もダメージを受けているoTargetの仲間（同じファクションのメンバー）の場所にジャンプする。目標が4 m以内にいる場合や、目標が見えていない場合はジャンプしない。（これによって別エリアにジャンプする危険がない）

これを使い、アンデッドの召集などに応用することができる。

Remarks

nw_c2_dimdoorはインクルードファイルではないため、通常の方法では使用できないが、コピーして使うことができる。

このスクリプトはブリンクドッグに使われている。使用方法はブリンクドッグのOnSpawnとOnUserDefinedスクリプトを参考にしよう。

Version

1.28

See Also

categories: [Combat Functions](#) | [Combat Actions Functions](#) | [Private Functions Functions](#)

author: Lilac Soul, additional contributor(s): Lilac Soul, JP team: ngtaicho

SearchForPlayer()

一番近いPCを探し攻撃する。

```
void SearchForPlayer();
```

Description

一番近いPCはGetNearestCreature命令で得られる。もしPCが見つからなかった場合は、頭をかき、30%の確立で1行のみの会話を発する。もしPCを発見した場合は攻撃を開始する。

Remarks

nw_c2_recruit1.nss の 16 行目

インクルードファイルではないため、通常の方法では使用できないが、コピーして使うことができる。

Version

1.28

See Also

categories: [Combat Functions](#) | [Combat Information Functions](#) | [Private Functions Functions](#)

author: Lilac Soul, JP team: ngtaicho

TalentRangedAttackers(object)

射程ある攻撃法で攻撃されている間の、呪文詠唱者であるNPCの行動方針を決定します。

```
int TalentRangedAttackers(  
    object oIntruder = OBJECT_INVALID  
);
```

Parameters

oIntruder
(デフォルト: OBJECT_INVALID)

Description

現在接近戦状態でない限り、射程ある攻撃者を狙います。現在の射程のある攻撃者が単独で、OBJECT_SELFとTargetが5メートル以上離れている場合、適切な範囲攻撃（指定）のためにOBJECT_SELFを調べます。

1以上の攻撃者がいた場合、範囲攻撃（無指定）能力を調べます。もし両方の範囲攻撃チェックが無効であった場合、遠距離攻撃能力をチェックします。

Requirements

```
#include "nw_io_generic"
```

Version

1.29

See Also

functions: [TalentMeleeAttack](#) | [TalentMeleeAttacked](#) | [TalentRangedEnemies](#) | [TalentSpellAttack](#)
categories: [Combat Functions](#) | [Core AI Talent Functions](#) | [Talents/Skills/Feats Functions](#)
events: [OnCombatRoundEnd Event](#) | [OnPhysicalAttacked Event](#)

author: Jody Fletcher, JP team: Rainie